1972

Identification of system dynamics using multiple integrations, tests of physical systems.

Montero, Angel Jose.

Monterey, California. Naval Postgraduate School

IDENTIFICATION OF SYSTEM DYNAMICS USING MULTIPLE INTEGRATIONS, TESTS OF PHYSICAL SYSTEMS

Angel Jose Montero

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

IDENTIFICATION OF SYSTEM DYNAMICS
USING MULTIPLE INTEGRATIONS,
TESTS OF PHYSICAL SYSTEMS

by

Angel Jose Montero

Thesis Advisor: George J. Thaler

June 1972

Approved for public release; distribution unlimited

IDENTIFICATION OF SYSTEM DYNAMICS

USING MULTIPLE INTEGRATIONS,

TESTS OF PHYSICAL SYSTEMS

by

Angel José Montero
Lieutenant, Spanish Navy
B.S., U.S. Naval Postgraduate School, 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1972

ABSTRACT

A method for identifying and - after identification - reducing linear time invariant systems on the basis of input - output records of the system is reviewed, applied to physical devices and extended to handle the case where noise is present. The method is implemented using an analog to digital converter in order to have the input and output records in a digitized form and thus to be able to use a digital computer program composed of a numerical integration subroutine and another subroutine to solve overdetermined sets of linear algebraic equations. Several practical examples are presented to demonstrate the accuracy and present capabilities of the procedure.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

The author would like to acknowledge the generous assistance given by Mr. Robert Limes in advising and programming on the Hybrid Computer System and especially to Dr. George J. Thaler of the Electrical Engineering Department of the Naval Postgraduate School for the guidance and helpful suggestions on the preparation of this research.

I. INTRODUCTION

A. THE IDENTIFICATION PROBLEM

All the modern techniques of analysis and design of any kind of system are based on the previous knowledge of a mathematical model of the system to be studied. This principle is also applicable to control systems. The form of the model depends on the methods of analysis and design to be employed as well as on the physical characteristics of the system and since most of the theory on control systems is based either on the state space or transform representation of systems, the majority of the mathematical models will be a set of state equations or a transfer function.

After deciding what basic form the mathematical model should take, the problem of determining the numerical values of the parameters arises. These values can often be determined from the laws of physics and the data supplied by the manufacturer or obtained through testing. This is not always the case, however; sometimes the laws of physics become mathematically intractable or are very difficult to apply and quite often the values of certain key parameters are not available or have changed with time. It is in these cases when the identification problem arises. This problem consists in determining the system model given the input, output and initial conditions.

For the purpose of this work, the identification problem can be stated as follows:

Given a system considered a black box, obtain a record of the input and output of the system over some finite period of time - 400 msecs. in this particular case - and from it find a mathematical model - the transfer function relating output and input in this case - and the values of the parameters in such a way that the model will accurately describe the system.

It should be kept in mind, however, that the problem of identifying a system solely on the basis of input and output quantities is very rare in engineering, because in most of the cases the engineer has some idea whether the system is linear or non-linear, time varying or time invariant, what is the order of the system itself and perhaps what are the time constants involved and in the presence of noise the statistical characteristics of it. For these reasons rather than having a BLACK BOX, what the engineers have is what could be called - a GREY BOX -, and as will be shown later, the identification techniques depend very heavily on knowledge of the characteristics mentioned above and in this particular case THE ORDER OF THE SYSTEM.

Although the control systems literature on system identification techniques is quite vast, there are no known identification schemes capable of handling all identification problems effectively. Some of the proposed identification methods are mentioned in the following:

Nieman, Fisher and Seborg [1] have pusblished a paper summarising most of the common approaches to the problem of identifying lumped parameter systems. Eykhoff [2] has a good discussion of the industrial application of several methods. Since the response to an impulse gives directly the transfer function of linear systems, obtaining this response is a common scheme used in linear identification. Mishkin and Haddad [3] have developed a technique for finding the impulse response based on samples of the system output and its derivatives. Levin [4] and Kerr and Surber [5] have developed a scheme for estimation of the impulse response from the samples of a noisy input and output. Turin [6] and Lichtenberger [7, 8] have used a matched filter to obtain an identification. Hill and McMurtry [9] have suggested the use of white noise, a binary test signal and crosscorrelation techniques to attack the problem. Cooper and Lindenlaub [10] have investigated the use of matched filter and crosscorrelation.

Another common approach to the problem is to determine the coefficients of the differential or difference equation which describes the system. Kumar and Sridhar [11] have employed the method of quasilinearization with some success. Nagumo and Noda [12] have developed a learning approach to the problem. Bass [13] uses modulating functions and the method works well in the presence of noise. Astron and Bohlin [14] use a statistical optimal method of determining differential equation parameters known as the "maximum likelihood method". Peterka and Smuk [15, 16] have developed

a similar method which is not optimal but is considerably simpler computationally and it is known as the "prior knowledge fitting method". Ho and Kalman [17] have proposed an algorithm for determining state variable models of sampled data systems which works well in the presence of noise, and has been extended to continuous systems by Eldem [18].

Identification methods vary widely with respect to how much must be known about the system before the method can be applied. In general, the less known about a given system, the more difficult it is to find a method capable of accomplishing the identification.

B. OBJECTIVES OF THIS PAPER

This paper will present a research, handling actual hardware, of an identification technique originally suggested by Diamesis [19] and studied using simulation techniques by W. R. Hansell [20]; also an attempt at using the technique in the presence of noise is made. The method requires a knowledge of the system input and output over some finite time interval and a rough estimate of the order of the system is desired. Although, theoretically the system input used need not be restricted to a class of testing functions, that is, it can be completely arbitrary, experimental results showed that inputs such as step, ramp or in general any periodic wave, sine, square, etc., should be avoided.

Unlike some identification schemes which require the calculation of derivatives of the input and output, the

technique to be presented requires only integrals of the input and output, and the advantages of numerical integration over differentiation are well known.

In Chapter II, the theoretical development of the identification technique is given. A method for identifying the initial conditions of the unknown system is also presented, but in this case it is necessary to know the value of the derivatives of the input at time zero. Since the initial conditions do not modify in any sense the mathematical model, their knowledge is not important.

Chapter III presents the method used to check the developed theory in Chapter II. Several examples are given to demonstrate the capabilities of the method.

In Chapter IV, recommendation and conclusions are given and following this chapter a complete listing of all computer programs are presented.

II. IDENTIFICATION BY MULTIPLE INTEGRATIONS

A. GENERAL APPROACH

The development that follows is similar to the ones given by Diamesis [19] and W. R. Hansell [20]. The development given by Diamesis is restricted to the case where all initial conditions are zero and the parameters of the model are the unknowns of a uniquely determined set of linear algebraic equations. Hansell extended the development to the case where the initial conditions need not be zero and he makes use of an overdetermined set of linear algebriac equations with the model parameters and some coefficients which have information about the initial conditions, as unknowns.

The overdetermined set of equations are solved using the method of least squares. The method is extended to handle noisy inputs and outputs.

Any single input, single output, lumped parameter, linear time invariant system can be described by a linear ordinary differential equation with constant coefficients. The general form of this equation is given in equation (1). A set of initial conditions is also necessary to completely define the system.

$$y^{n)}(t) + a_{n-1}y^{n-1)}(t) + \ldots\ldots\ldots\ldots + a_o y(t) = \tag{1}$$

$= b_m U^{m)}(t) + \ldots\ldots\ldots\ldots + b_o U(t)$ where, in a general case $n>m$ and

$y^{i)}(t)$ and $U^{j)}(t)$ represent the i^{th} and j^{th} derivatives of the output and input, respectively.

The necessary initial conditions are:

The value at time zero, of the output and its n-1 first derivatives and the value at time zero, of the input and its m-1 first derivatives.

The notation to be used is:

$y(o), y^{1)}(o) \ldots\ldots\ldots\ldots y^{n-2)}(o), y^{n-1)}(o)$ to denote the, n-1, I.C. of the output and

$u(o), u^{1)}(o) \ldots\ldots\ldots\ldots u^{m-2}(o), u^{m-1}(o)$ to denote the, m-1, I.C. of the input.

The identification problem to be treated here consists of determining the coefficients a_i and b_j of equation (1), on the basis of input and output records taken over some arbitrary time interval, for every pair of values chosen for n and m. The initial conditions will be assumed to be unknown and also it is assumed that the noise present is zero mean and that the standard deviation is small compared with the R.M.S. of the input and output, for the technique to work properly.

Taking the Laplace transform of (1), yields equation (2)

$$s^nY(s) + a_{n-1}s^{n-1}Y(s) + \ldots\ldots\ldots\ldots + a_oY(s) =$$
$$b_ms^mU(s) + \ldots\ldots\ldots\ldots + b_oU(s) + d_{n-1}s^{n-1} + \ldots + d_o \quad (2)$$

where the d_k coefficients account for the contribution of the initial conditions.

Dividing equation (2) by s^{n+1} results in equation (3)

$$\frac{Y(s)}{s} + a_{n-1}\frac{Y(s)}{s^2} + \ldots\ldots\ldots\ldots + a_o\frac{Y(s)}{s^{n+1}} = b_m\frac{U(s)}{s^{n-m+1}} + \ldots + b_o\frac{U(s)}{s^{n+1}} + d_{n-1}\frac{1}{s^2} + \ldots\ldots\ldots\ldots + d_o\frac{1}{s^{n+1}} \qquad (3)$$

Taking now the inverse transform of equation (3), gives equation (4).

$$\int_o^{t_f} y(t)dt + a_{n-1}\int_o^{t_f}\int_o^{t} y(t)dt^2 + \ldots + \ldots + a_o\int_o^{t_f} \ldots(n+1)\ldots \int_o^{t} y(t)dt^{n+1} =$$

$$= b_m\int_o^{t_f} \ldots\ (n-m+1)\ \ldots \int_o^{t} u(t)dt^{n-m+1} + \ldots + b_o\int_o^{t_f} \ldots\ (n+1)\ \ldots \int_o^{t} u(t)dt^{n+1} + \qquad (4)$$

$$d_{n-1}\int_o^{t_f}\int_o^{t} dt^2 + \ldots + d_o\int_o^{t_f} \ldots\ (n+1)\ \ldots \int_o^{t} dt^{n+1}$$

And since the system is assumed to be time invariant, nothing has been lost by setting the lower limit on the integrals equal to zero.

Rearranging terms in equation (4), equation (5) is obtained.

$$a_o \int_o^{t_f} \cdots (n+1) \cdots \int_o^{t} y(t)dt^{n+1} + \cdots$$

$$+ a_{n-1} \int_o^{t_f} \int_o^{t} y(t)dt^2 + \cdots\cdots\cdots\cdots\cdots\cdots$$

$$- b_o \int_o^{t_f} \cdots (n+1) \cdots \int_o^{t} u(t)dt^{n+1} + \cdots -$$

$$b_m \int_o^{t_f} \cdots (n-m+1) \cdots \int_o^{t} u(t)dt^{n-m+1} - \qquad (5)$$

$$d_o \int_o^{t_f} \cdots (n+1) \cdots \int_o^{t} dt^{n+1} - \cdots -$$

$$d_{n-1} \int_o^{t_f} \int_o^{t} dt^2 = - \int_o^{t_f} y(t)dt$$

Since records of the input and output are assumed to be known, a linear algebraic equation with the system parameters a_i, b_j, and the terms that contain information about the initial conditions as unknowns, can be formulated by performing the multiple integrations, from zero to some time t_f, or from any time t_o to a final time t_f, after assuming some values for n, and m. A set of 2n+m+1 equations can be obtained by

letting t_f take on 2n+m+1 different values. Assuming that the equations are linearly independent, it will now be possible to solve for the 2n+m+1 unknowns; n+m+1, a_i, b_j coefficients and the n d_k terms which contain information about the initial conditions. It will be shown later how those coefficients can be used to determine the initial conditions.

Consider now the case when noise is present at the input, at the output, or at both.

Under these conditions equation (5) becomes:

$$a_o \int_o^{t_f} \dots (n+1) \dots \int_o^{t} \left[y(t)+v(t)\right] dt^{n+1} +$$

$$\dots + a_{n-1} \int_o^{t_f} \int_o^{t} \left[y(t)+v(t)\right] dt^2 -$$

$$b_o \int_o^{t_f} \dots n+1 \dots \int_o^{t} \left[u(t)+r(t)\right] dt^{n+1} -$$

$$\dots - b_m \int_o^{t_f} \dots n-m+1) \dots \int_o^{t} \left[u(t)+r(t)\right] dt^{n-m+1} \qquad (6)$$

$$d_o \int_o^{t_f} \dots n+1 \dots \int_o^{t} dt^{n+1} - \dots\dots\dots -$$

$$d_{n-1} \int_o^{t_f} \int_o^{t} dt^2 = - \int_o^{t_f} y(t)+v(t) \quad dt$$

where v(t) and r(t) represent the noise at the output and input respectively.

If the coefficients in equation (6), that is, the multiple integrals of the input, output, and time, are going to have the same values as the corresponding ones in equation (5), in order to have the same solution for the a_i, b_j, and d_k, in other words, the same identification, the presence of noise should not affect the value of the integrals.

Consider a general term:

$$\int_o^{t_f} \cdots (p) \cdots \int_o^{t} \left[y(t)+v(t)\right] dt^p \tag{7}$$

Equation (7) can be written:

$$\int_o^{t_f} \cdots (p) \cdots \int_o^{t} y(t)dt^p + \int_o^{t_f} \cdots (p) .. \int_o^{t} v(t)dt^p$$

The first integral of equation (8). (8)

$$\int_o^{t_f} \cdots (p) \cdots \int_o^{t} y(t)dt^p \tag{9}$$

is the true value of the coefficient of a_{n-p+1} in equation (5) and

$$\int_{o}^{t_f} \cdots (p) \cdots \int_{o}^{t} v(t)dt^p \tag{10}$$

the undesired contribution of the noise. Thus, what has to be done is to find when the value given by equation (10) is negligible with respect to the value of equation (9).

Assuming zero mean noise, the expected value of equation (10) is:

$$E\left[\int_{o}^{t_f} \cdots (p) \cdots \int_{o}^{t} v(t)dt^p\right] =$$

$$\int_{o}^{t_f} \cdots (p) \cdots \int_{o}^{t} E\left[v(t)\right] dt^p = 0 \tag{11}$$

Since zero mean noise was assumed and the expected value of the integral is equal to the integral of the mean value. But the fact that the mean value of equation (10) is zero is not enough to say that the influence of the value of equation (10) in (7) is negligible. Besides the mean of equation (10), its standard deviation or R.M.S. value should be found in order to be able to compare it with equation (9), and if the R.M.S. value of equation (7) is very small compared with the value of equation (9), it can be said that the influence of the noise will not significantly affect the identification.

The variance of equation (10) considering, p = 2, for simplicity, is given by equation (12)

$$\sigma_z^2 = E\left[\left(\int_o^t \int_o^\xi v(\gamma)\,d\gamma\,d\xi\right)^2\right] \tag{12}$$

where γ and ξ are dummy variables. Equation (12) can be written

$$\sigma_z^2 = E\left[\int_o^t \int_o^{\xi_1} v(\gamma_1)\,d\gamma_1\,d\xi_1 \int_o^t \int_o^{\xi_2} v(\gamma_2)\,d\gamma_2\,d\xi_2\right] =$$

$$= \int_o^t \int_o^t \int_o^{\xi_1} \int_o^{\xi_2} E\left[v(\gamma_1)\ v(\gamma_2)\right] d\gamma_1\,d\gamma_2\,d\xi_1\,d\xi_2 \tag{13}$$

but $E\left[v(\gamma_1)v(\gamma_2)\right]$ is recognized to be the auto-correlation and can be written as $R_v(\tau)$, where $\tau = \gamma_2 - \gamma_1$.

The R.M.S. value of equation (10) can now be expressed as follows:

$$\sigma_z = \left(\int_o^t \int_o^t \int_o^{\xi_1} \int_o^{\xi_2} R_v(\tau)\,d\gamma_1 d\gamma_2 d\xi_1 d\xi_2\right)^{\frac{1}{2}} \leq$$

$$\left(\int_o^t \int_o^t \int_o^{\xi_1} \int_o^{\xi_2} R_v(o)\,d\gamma_1 d\gamma_2 d\xi_1 d\xi_2\right)^{\frac{1}{2}} =$$

$$= \left(\int_o^t \int_o^{\xi_1} \sqrt{R_v(o)}\,d\gamma_1\,d\xi_1 \int_o^t \int_o^{\xi_2} \sqrt{R_v(o)}\,d\gamma_2 d\xi_2\right)^{\frac{1}{2}} =$$

$$= \int_o^t \int_o^\xi \sqrt{R_v(o)}\,d\gamma\,d\xi \quad ; \quad R_v(\tau) \leq R_v(o) \tag{13a}$$

or going back to the initial notation

$$\sigma_z = \int_o^{t_f} \int_o^{t} \sqrt{R_v(o)}\, dt^2 \tag{14}$$

If now (14) is compared with (9), with p = 2, it is observed that both integrals have the same form. Therefore, if

$y(t) >> R_v(o)$ = R.M.S. value of v(t), the noise is not going to have much influence in the identification. The same conclusion can be reached for any value of p.

As was said before, the set of 2n+m+1 linearly independent algebraic equations obtained from equation (5) letting t_f take 2n+m+1 different values, would solve the problem of identifying the parameters a_i, b_j, and d_k, but since for some kind of inputs and some t_f, the 2n+m+1 equations may not be linearly independent, thus instead of forming just 2n+m+1 equations, a set is formed which has a number of equations substantially bigger than 2n+m+1 and this over-determined set is solved using the method of least squares.

This technique also allows some flexibility in the selection of the initial values of n and m as will be shown later.

If in equation (2) the initial conditions are set equal to zero, after a simple manipulation, the transfer function of the system can be obtained and is given in equation (15).

$$\frac{Y(s)}{U(s)} = \frac{b_m s^m + \ldots + b_1 s + b_o}{s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_o} \tag{15}$$

It should be noticed that the coefficients a_i and b_j of equation (10) are exactly the same as the a_i and b_j of equation (5). Therefore, as soon as the set of overdetermined algebraic equations is solved for a_i and b_j, the transfer function is determined.

Equation (15) is usually written in the form (16).

$$\frac{Y(s)}{U(s)} = \frac{K\left[s^m + c_{m-1}s^{m-1} + \ldots + c_1 s + c_o\right]}{s^n + a_{n-1}s^{m-1} + \ldots + a_1 s + a_o} \qquad (16)$$

Where $K = b_m$ is the so called constant gain and $c_g = \frac{b_g}{b_m}$ for $o \leq g \leq m$.

If the roots of the numerator and denominator polynomials are found, the more familiar form of the transfer function can be written; equation (17).

$$\frac{Y(s)}{U(s)} = K \frac{\prod_{j=1}^{m} (s + q_j)}{\prod_{i=1}^{n} (s + p_i)} \qquad (17)$$

Where $-p_i$ and $-q_j$, are the poles and zeros respectively.

Equation (16) can also very easily be written in state form as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \left[\begin{array}{c|ccc} 0 & 1 & & \\ 0 & 0 & \ddots & \\ \vdots & & & \\ 0 & 0 & & 1 \\ \hline -a_o & -a_1 & \cdots & -1 \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ K \end{bmatrix} \begin{bmatrix} u \end{bmatrix}$$

$$y = \begin{bmatrix} c_o & c_1 & \cdot \cdot \cdot \cdot & 1 & 0 & \cdot \cdot \cdot \cdot & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \tag{17a}$$

But, how are n and m determined? It was said that in order to apply the method efficiently a rough idea of the order of the system should be available, although it is not absolutely necessary; that rough idea is used to pick n' and m' as the first approximation for n and m, taking into account that n'>n and m'>m to be sure that neither poles nor zeros are missing. Since m is usually smaller than n, m' should not be greater than n'-1.

Consider then that n and m are the true numbers of poles and zeros and that n' and m' are assumed. Under these conditions one would expect to obtain n'-n, or m'-m - whatever number is smaller - poles equal to zeros, what is equivalent to saying that they are not present in the system, what is actually the case; the (n' - n) - (m' - m) poles or (m'- n) - (n' - n) zeros in excess which have not been cancelled will appear to be either in the right half plane or with values too big compared with the actual values of the poles and zeros in the system. As a result after a first guessing for n and m, a clear indication of the actual order of the system and of the number of zeros, should show up. Examples in Chapter III will demonstrate this point.

B. IDENTIFICATION OF THE INITIAL CONDITIONS

Although the knowledge of the initial conditions is not important by itself, it might be if one desired to compare the system model with the actual system by exciting the system model with the same input that was used in the identification. By comparing the output of the model with the output of the system a rough idea of the accuracy of the model can be obtained. However, in order for this to be possible, all the initial conditions of the input and output should be known.

When the overdetermined set of linear algebraic equations is solved, besides the values of a_i and b_j, the d_k initial condition terms are also found. These d_k terms are related with the initial conditions by equation (18). This equation can easily be found by specifically writing in the contribution of the individual initial conditions when the Leplace Transform of equation (1) was taken, to obtain equation (2).

$$\begin{bmatrix} d_o \\ d_1 \\ \cdot \\ \cdot \\ \cdot \\ d_{n-2} \\ d_{n-1} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-1} & 1 \\ a_2 & a_3 & & \diagup & \\ \vdots & & \diagup & & \\ a_{n-1} & \diagup & & 0 & \\ 1 & & & & \end{bmatrix} \times \begin{bmatrix} y(o) \\ y^{1)}(o) \\ \cdot \\ \cdot \\ \cdot \\ y^{n-2)}(o) \\ y^{n-1)}(o) \end{bmatrix} - \tag{18}$$

$$
-K\left[\begin{array}{cccccc|ccc}
b_1 & b_2 & \cdots & \cdots & b_{m-1} & 1 & 0 & \cdots & 0 \\
b_2 & b_3 & & & 1 & & \cdot & & \cdot \\
\cdot & & & \cdot^{\cdot^{\cdot}} & & & \cdot & & \cdot \\
\cdot & & & & 0 & & \cdot & & \cdot \\
\cdot & & & & & & \cdot & & \cdot \\
b_{m-1} & 1 & & & & & \cdot & & \cdot \\
1 & & & & & & 0 & \cdots & 0 \\
\hline
0 & & & & & & 0 & \cdots & 0 \\
\cdot & & & 0 & & & \cdot & & \cdot \\
\cdot & & & & & & \cdot & & \cdot \\
0 & & & & & & 0 & \cdots & 0
\end{array}\right]
\times
\left[\begin{array}{c}
u(o) \\
u^{1)}(o) \\
\cdot \\
\cdot \\
\cdot \\
u^{m-1}(o) \\
\hline
0 \\
\cdot \\
\cdot \\
0
\end{array}\right]
\qquad (18)
$$

Equation (18) requires the knowledge of m-1 derivatives of the input and numerical techniques may be necessary to find them, however, this difficulty may be avoided in many cases by choosing the beginning of the input-output record at a moment where the input is constant and thus all the derivatives would be zero.

C. SPECIAL CASES WHERE SIMPLIFICATION COULD BE POSSIBLE

Theoretically the method should identify the system independently of the number of unknowns,but practically this is not the case due to the natural limitations in the accuracy of the numerical integration methods and in the measure of the input and output records, thus it would be desirable to reduce the number of unknowns as much as possible, because as a general rule the fewer the unknowns, the more accurate its identification.

A great reduction in the number of unknowns occurs if the record of the input and output begins when all the initial conditions are zero. Under these conditions, all

the terms d_k of equation (5) are zero and the number of unknowns is reduced from 2n+m+1 to n+m+1 since it was assumed n>m.

Any additional information could be used to simplify the problem, for instance if the value of the steady state gain constant were known the number of unknowns could be reduced by one.

III. IMPLEMENTATION

A. NUMERICAL METHODS

As was mentioned before, after having the record of the input and output of the system, which can be obtained as described in Appendix I, it is necessary to perform the multiple integrations indicated in equation (5), for different times t_f, in order to form the set of overdetermined linear algebraic equations whose solution gives the values of the parameters a_i, b_j and d_k, although only the a_i and b_j are of interest in this work. The d_k, even available are not used.

The multiple integrations can be handled for any numerical method subroutine and although there are quite a few highly sophisticated numerical integration procedures available, trapezoidal integration will be used because most of the more complex integration techniques perform poorly when the function being integrated is discontinuous. Since the inputs used here are step-wise waves, complex integration techniques are not desirable. On the other hand, the slight improvement in accuracy offered by more advanced methods is not, in general, enough to justify the tremendous increase in computational load associated with their use.

The solution of the set of overdetermined linear equations is a classical problem in several fields of mathematics and engineering. Most of the classical techniques for solving

such problem are not practical. They tend to magnify the errors introduced by the finite precision of the computer to the point where the solution is meaningless. Fortunately, several modern methods are available that display more acceptable behaviour. The method used here was developed by Golub [22]. The basic approach is to triangularize the coefficient matrix by performing a Choleski decomposition. The decomposition is accomplished by applying repeated Householder transformations [23]. Once the coefficient matrix has been triangularized the unknowns can be obtained by back substitution. The method is quite stable numerically and is capable of handling ill-conditioned coefficient matrices.

B. CHARACTERISTICS OF GOOD INPUT-OUTPUT RECORDS

The accuracy with which a system can be identified is strongly dependent on the input-output record used in the identification. Since parameters are identified on the basis of their effect on the output, it will be impossible to identify a parameter unless its effect is measurable. If the input to a system has a frequency spectrum that is more or less uniform over the frequency range of interest, the identification will probably be very good. If the frequency spectrum of the input is confined to a narrow band of frequencies, the identification will probably be very bad. It is well known that signals with sharp discontinuities have a broader bandwidth than slowly varying continuous signals.

For this reason, the step-wise wave was chosen as the input in this research, after having demonstrated experimentally the preceeding observations seeing that with a step input it was never possible to obtain a good identification even for a second order system. Periodic functions should also be avoided at the input, because very probably many of the algebraic equations are going to be linearly dependent. This fact was also proved experimentally; a periodic square wave was tried as input and the identification was also bad when a step-wise input gave very good results.

Another desired characteristics of the input, experimentally proved, is that it should be adjusted in such a way that the output, if it is strictly increasing, should at some moment in the sampling interval have a negative second derivative; in other words, the output is desired to have an inflexion point within the period in which the record is taken. The reason for this is thought to lie in the subroutine DLLSQ which solves the overdetermined set of algebraic equations. What DLLSQ essentially does is look for a fitting output function that minimizes the least square error between the generated and the measured and if the second and higher order derivatives of the output are positive at the end of the sampling interval, the function generated by the subroutine may in some cases correspond to an unstable situation. Examples in Chapter IV will also demonstrate this point.
In relation to the sampling rate at which the input and output should be digitized, it can be said that low sampling rates

may introduce inaccuracies in the numerical integrations and as a result poor identification, especially in the location of high frequency poles and zeros. Maximum sampling rate available or ten to one hundred times shorter than the shortest system time constant is recommended. The output of the system is also recommended to have maximum value of the order of the maximum value of the input, thus the numbers with which the computer deals are not very different in magnitude. If the output has not the same order of magnitude of the input, it can as a rule be amplified accordingly, taking this amplification factor into account when the time comes to see what the actual identification is.

C. PROGRAMS DESCRIPTION

Four different computer programs were used in this research. A listing of them is given after Chapter IV. The first one is the A/D program. This program is used for high speed A/D (analog to digital) conversion. Its output is a seven track magnetic tape that contains the digitized input and output record of the system. This program consists of a main driver program and five subroutines.

The source language is FORTRAN for the main program and METASYMBOL for the subroutines.

The A/D program provides a means of sampling and digitizing analog information at a rate (up to 10KHz) determined by an interrupt signal originated at the analog computer logic patchboard (Ci 5000). (See Appendix I). The digital data

are stored sequentially in one of two buffers provided by the main program. When one buffer is full, the storage sequence is continued at the top of the other buffer. An indicator notifies the main program of this switching and the main program processes the data in the full buffer writing them in the magnetic tape while the other buffer is being filled.

The parameters to be entered in the main program are:

NSAMP = Number of words in one block/NCHAN.

NCHAN = Number of channels to be digitized.

NREC = Maximum number of records to be digitized in one file.

ITAPE = Number of tape drive unit to be used.

NDEL = Delay between digital and analog signal in option 7. The options of this program are:

1 = Desired to input new parameters.

2 = Start digitizing the analog input signals.

3 = Write an END OF FILE on the magnetic tape.

4 = Rewind the tape to the loading point (beginning).

5 = Skip the following number of END OF FILE marks.

6 = Print the following amount of digitized data.

7 = Actuate the A/D program for the next single block of data encountered and feed the signal to the recorder.

The five subroutines are: ADSTART, ADSTOP, ADFAST, MTRDY and MTOUT.

Subroutine ADSTART initializes the A/D input operations. The calling sequence is:

CALL ADSTART (NCH, BUF, NEWBUF, NSAM, RECNUM, NEXLOC),

where

NCH = Number of A/D channels to sample.

BUF = A pointer to the first buffer.

NEWBUF = A pointer to the second buffer.

NSAM = Number of samples per buffer.

RECNUM = A record counter which is bumped by ADFAST each time a buffer is filled.

NEXLOC = The return address for ADFAST to be used each time a buffer is filled.

The first four arguments are used to set up pointers and counters in ADFAST. RECNUM provides a means for ADFAST to keep the main program aware of the record number of the buffer currently being filled. NEXLOC is an alternate return location for ADFAST. ADSTART also sets up the interrupt controller branch instructions and starts the A/D hardware. ADSTOP stops the A/D operations and restores the interrupt locations to their normal instructions.

ADFAST controls the buffer location pointer which directs the A/D data into core.

MTRDY initializes the magnetic tape handler. It has the following arguments:

UNIT = Magnetic tape unit number.

BUF1 = First word address of first buffer.

BUF2 = First word address of second buffer.

NW = Number of words to be written per record.

IR = Tape operation status word return.

1 if tape I/O channel is busy.

2 if operation completed.

3 if tape unit is not ready.

4 if tape writes error.

MTOUT controls the stream of A/D data from the buffers into the tape. It is called by the main program each time a record of data is ready for recording. The argument, NB, indicates which buffer is just filled and is to be output. After setting the status indicator (IR) to 1, to indicate an operation not complete condition, the routine tests the tape and channel status. If either is busy, IR is bumped to 3 (not ready condition) and the routine exits back to the calling program. If everything is all right, NB is checked and the I/O instructions for writing the filled buffer on tape are executed. If the operation is completed without error, IR is set to 2 (operation completed) and to 4 (write error) if an error occurred, and execution is returned to main program.

The seven track magnetic tape generated by the A/D program recording the digitized input and output can not be directly read by the IBM 360 computer system, and since this system is used to perform the integrations of equation (5), a seven to nine track tape conversion should be done prior to the numerical integration. This conversion is done by the IBM 360 using the CONVERT program, which is described in Ref. 21. This program has several parameters. One of them is BLKSIZE which is related with the NSAMP and NCHAN

parameters used in the A/D program. BLKSIZE can not be greater than 32.700 and for NCHAN = 2 limits the maximum number of samples that can be taken for the input and output record to about 4000.

The third program is the main one and is a direct implementation of the procedure developed in Chapter II. This program has several important parameters that have to be defined at the very beginning. NP and NZ are rough estimates of the number of poles and zeros in the system to be identified. NP and NZ stand for n' and m' in the program, N' and m' were defined in Chapter II. KPTMAX is the number of sample points available in the input-output record. The input amplitude is R and the output amplitude is C. Because the integration of the time has to be done, the program generates the time corresponding to each sample for what the sampling rate has to be known and is represented by the parameter DT in the program. IPTS is the number of sample points that will separate successive linear equations. Therefore, the total number of linear algebraic equations formed is equal to KPTMAX/IPTS. The program reads in the nine track tape generated by the CONVERT program, performs the integrations of equation (5) and forms the overdetermined set of linear algebraic equations. When the set is formed, subroutine DLLSQ is called. This subroutine is an implementation of the Golub algorithm for solving overdetermined sets of linear equations. Subroutine DLLSQ returns the values a_i, b_j and d_k. The a_i and b_j coefficients are fed

into the RTPLSB subroutine, which is a polynomial root finder, in order to obtain the poles and zeros of the system.

The output of the program are the results of the identification given in both transfer function and state variable form.

The fourth program was written to compare the outputs of the actual system and of the ones obtained from the identification, when all of them are excited with a step. This program solves the differential equations that describe the system. The differentials equations should be arranged in state variable form. The output of the program is a drawing of the response of the different systems to be compared in the same plot.

D. EXAMPLES

The following examples show the possibilities of the method, how the order of the system can in general be determined, if it is not already known, from a trial identification run using an estimated order greater than the actual order of the system. They also show how the accuracy of an identification depends on the input wave form, on the R.M.S. value of the noise present, and as expected, does not depend on the initial conditions. They also indicate how the method can be used to reduce or obtain a lower order mathematical model. In all the examples, the MAIN PROGRAM PARAMETERS were:

NUMBER OF DATA POINTS = 4000

NUMBER OF EQUATIONS = 40

DATA POINTS PER EQUATION = 100

In examples 1 through 6, the input function was a step-wise increasing wave form (staircase wave form, see Fig. 1, Example 1), with a maximum value of about 85 volts. The amplitude and the number of steps were changed in each example. In Example 3 and 4, the maximum amplitude of the input was about 9 volts. In examples 6 through 10, the input function was also a step-wise wave form, but instead of being strictly increasing, it had one part strictly increasing and the final part strictly decreasing. See Fig. 4, Example 5.

The systems investigated in Examples 1, 2, 5, 6, 7, 8, 9, 10 and 11 were set up using R-C and R-C-L circuits isolated by operational amplifiers which also provided the desired amplification factor. In Examples 3 and 4, an active system was studied. This system was a D.C. positioning servo.

Example 1 was performed as a first test and in it a second order system - two real poles - was considered. The input was a step-wise increasing function, (see Fig. 1), with zero initial conditions. The results of the identification were within the 2% for the dominant pole and the gain constant and within the 10% for the higher frequency pole. This example shows how assuming different values for m' and n', the results of the identification gave a clear indication of what the actual order of the system was.

Example 2 illustrates the identification of a third order system. The input function as in Example 1 was a step-wise increasing function. The initial conditions were also zero. The identification of the low frequency poles and the gain constant were also within 3% of the true values; however, the pole at -100.0, was identified with an error of about 30%, this result was not a surprise since the more negative the value of a pole, compared with the other poles in the system, the less exact its identification. This example shows also the fact that in some cases the trial runs to find out the true values of n' and m', did not indicate very clearly what these values were. In cases like this, a thorough search should be done around the expected values of n and m and different input-output records are recommended to be used.

This example also shows how the method could be used to reduce the order of the system, i.e., when n' = 3 and m' = 1 were assumed, the results correspond practically to a second order system, since the pole at -2.22 and the zero at -2.20 cancel each other. When n' = 2 and m' = 0 were assumed, practically the same results were obtained. The steady state error coefficient of the second order has practically the same value of the third order. Essentially, the method neglected the effects of the higher frequency pole.

The experiment described in Example 3 was performed using a D.C. positioning servo. The transfer function of

the servo was first obtained from the asymptotic approximation of the closed loop frequency response plot shown in Fig. 2. The parameters of the servo were adjusted in such a way that two dominant complex roots were present. During the frequency response test, it was found that the servo had nonlinearities since a jump in the resonance frequency was observed and also that reducing the amplitude of the input sine wave from twenty volts to one volt peak to peak the resonance frequency was shifted from 2.6Hz to 8.1Hz (52 rad/sec).

In taking the input-output record in this example, the input was adjusted to be a step-wise increasing function, (see Fig. 1), but with a step amplitude of about 0.5 volts. The 4 volts step amplitude used in Examples 1 and 2 was too big; the D.C. motor ran continuously.

The results of the identification were almost identical to the ones obtained from the frequency response plot.

In Example 4, the same D.C. servo was treated but now the parameters of the servo were adjusted in order to have two real poles. Also a frequency response test was performed and a transfer function was obtained from the frequency response curve asymptotic approximation (see Fig. 3). The results of the identification were very close to the ones found from the frequency response plot.

In Example 5, a third order system is again considered, but now a zero was added.

The input function and the I.C. were the same as in Example 1. The results in this case also showed a poor identification of the higher frequency pole.

Example 6 shows the effect of the input function in the accuracy of the identification. Now a fourth order system is studied using two different inputs. The first input was the same as in Example 1 and is shown together with the output of the system in Fig. 4. With this input, the results of the identification were very poor since not a single pole was correctly identified. Several fourth order systems were treated with this type of input and the identification was bad, giving in some cases poles in the right half plane. It was observed, however, that the output of the system, at the end of the record had a positive second derivative (see Fig. 4). It was thought that perhaps this fact could be the cause of the poor identification considering that an unstable system might have the same output, for the finite interval of time considered, and this unstable system could be the one identified or generated by the DLLSQ subroutine. With the preceding in mind, the input function was changed and adjusted in such a way that the output had the second derivative negative at a certain instant within the time interval considered (see Fig. 5). The results in this case were very good since even the high frequency pole was identified with an error within the 6% of the true value. The I.C. were also zero value.

In Example 7, the identification of a system with four poles and two zeroes is presented. The input function was as the one shown in Fig. 5. The results in this case were also very good.

Example 8 shows how the I.C. does not affect the identification, the results being practically the same with I.C. equal to zero as with I.C. 0; several different initial conditions were considered.

In Example 9, a third order system was studied, but noise was added at the input and output. The results showed the great influence of the noise in the identification. It was found that in order to have a good identification, the maximum R.M.S. value of the noise should not be greater than about .5 volts. 1.0 volts R.M.S. noise made the identification very bad. The noise added was a gaussian, low frequency noise - from a few cycles per second to about 300 cycles per second - with zero mean.

Examples 10 and 11 show how the method can be used to reduce the order of the system. In Example 10, a fifth order system was treated. The identification was not good for the higher frequency poles. Other fifth order systems were studied and in some cases none of the poles was identified, which indicates that fourth order systems are perhaps the highest order systems that can be identified with the kind of inputs used in this work.

In Example 10, the reduction was not, as a direct consequence of the poor identification, very good as can be seen

comparing the responses to a step input of the different models, see Fig. 6. However, it is observed that the three models generated by the method, 5^{th}, 3^{rd}, and 2^{nd} order, give responses fairly close to one another.

In Example 11, although the identification is considered very good, the responses of the different reduced models practically coincide during the transient period, but in steady state an error of about 30% is observed between the step responses of the fourth order identified model and the third and second order reduced models (see Fig. 7). However, if the gain constant of the reduced models is adjusted in order to have the same steady state gain as the fourth order model, the responses are practically the same (see Fig. 8).

It is to be noticed that the systems considered true, in the examples, were true within the tolerances of the values of the resistors, capacitors, coils and amplifiers used.

EXAMPLE I

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-10.0	0.0	J
2	-100.0	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 1000.0

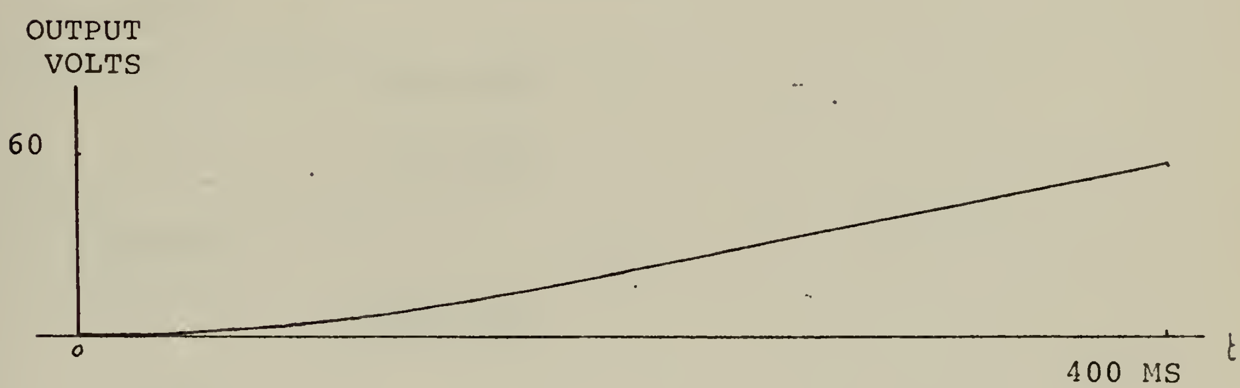
TESTS FOR EXAMPLE I

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	2	0	Good
b	3	1	See Table of Data
c	4	2	See Table of Data
d	2	1	See Table of Data

INPUT FUNCTION STEP-WISE STRICTLY INCREASING SHOWN IN FIG. 1.

INITIAL CONDITIONS EQUAL ZERO.

INPUT FUNCTION USED IN EXAMPLE NO. 1



OUTPUT FUNCTION EXAMPLE NO.1

FIGURE 1

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-9.8488868	0.0	J
2	-111.04012	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 984.74146

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	1093.6215
2	120.88900

B VECTOR

2	984.74146

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

GOOD RESULTS.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-97.994740	0.0	J
2	2.2362954	0.0	J
3	-9.8646060	0.0	J

ZEROES	REAL	IMAGINARY	
1	2.2427425	0.0	J

GAIN CONSTANT = 867.90796

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	-2161.7809
2	725.47414
3	105.62305

B VECTOR

3	867.90796

C VECTOR

1	-2.2427425
2	1.0000000

REMARKS:

ASK FOR THREE POLES ONE ZERO.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM SINCE ZERO AT 2.24 CANCELS POLE AT 2.23.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-4.4110354	-54.484437	J
2	-4.4110354	54.484437	J
3	-99.827986	0.0	J
4	-9.8612480	0.0	J

ZEROES	REAL	IMAGINARY	
1	-7.0865674	-59.922949	J
2	-7.0865674	59.922949	J

GAIN CONSTANT = 727.27368

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	2941483.4
2	336437.35
3	4940.1259
4	118.51130

B VECTOR

4	727.27368

C VECTOR

1	3640.9793
2	14.173135
3	1.0000000

REMARKS:

ASK FOR 4 POLES TWO ZEROES.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE COMPLEZ ZEROES CANCEL COMPLEX POLES.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (d)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-9.8592476	0.0	J
2	-97.619069	0.0	J

ZEROES	REAL	IMAGINARY	
1	-847.73348	0.0	J

GAIN CONSTANT = 1.0221453

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	962.47832
2	107.47832

B VECTOR

2	1.0221463

C VECTOR

1	847.73348
2	1.0000000

REMARKS:

ASK FOR TWO POLES ONE ZERO.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM. SINCE THE ZERO AT - 847.7 CAN BE NEGLECTED.

EXAMPLE 2

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.000	0.0	J
2	-10.00	0.0	J
3	-100.0	0.0	J

ZEROES

GAIN CONSTANT - 10000.0

TESTS FOR EXAMPLE 2

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	3	0	Good
b	4	1	See Table of Data
c	4	0	See Table of Data
d	4	2	See Table of Data
e	3	1	See Table of Data
f	3	1	See Table of Data

INPUT FUNCTION STEP-WISE STRICTLY INCREASED SIMILAR TO THE ONE SHOWN IN FIG. 1.

INITIAL CONDITIONS EQUAL ZERO.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.1072321	0.0	J
2	-10.224865	0.0	J
3	-131.11147	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 11153.488

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	1484.3522
2	1497.0893
3	142.44357

B VECTOR

3	11153.488

C VECTOR 1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

GOOD RESULTS FOR LOW FREQUENCY POLES, NOT SO GOOD FOR HIGH FREQUENCY POLE.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.1675351	0.0	J
2	-9.9863699	0.0	J
3	-1.7691065	-59.923677	J

ZEROES	REAL	IMAGINARY	
1	157.78387	0.0	J

GAIN CONSTANT = -1919.7240

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	41903.747
2	40128.130
3	3645.1011
4	14.692118

B VECTOR

4	-1919.1240

C VECTOR

1	-157.78387
2	1.0000000

REMARKS:

ASK FOR FOUR POLES ONE ZERO.

RESULTS TO NOT GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, INDICATING FOUR POLES WHEN THERE ARE THREE. THE

ZERO IN THE RIGHT HALF PLANE SHOULD BE NEGLECTED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.1300231	0.0	J
2	-10.158884	0.0	J
3	-11.337775	-53.285012	J
4	-11.337775	+53.285012	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 252014.00

SYSTEM STATE VARIABLE (PHASE FORM)

A VECTOR

1	34070.104
2	33763.954
3	3235.2996
4	33.964457

B VECTOR

4	252014.00

C VECTOR

1	1.0000000

REMARKS:

ASK FOR FOUR POLES NO ZEROES.

RESULTS DO NOT GIVE CLEAR INDICATION OF ORDER OF THE SYSTEM INDICATING FOUR POLES WHEN THERE ARE THREE.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (d)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.1196795	0.0	J
2	-10.190750	0.0	J
3	0.16021282	-59.917890	J
4	0.16021282	59.917890	J

ZEROES	REAL	IMAGINARY	
1	3.5557973	-30.108372	J
2	3.5557973	30.108372	J

GAIN CONSTANT = 332.37549

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	40965.288
2	40602.813
3	3597.9654
4	10.990004

B VECTOR

4	332.37549

C VECTOR

1	919.15775
2	-7.1115946
3	1.00000000

REMARKS:

ASK FOR FOUR POLES TWO ZEROES

RESULTS INDICATE TWO POLES WHEN THERE ARE THREE.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (e)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0459484	0.0	J
2	-10.251872	0.0	J
3	2.2244020	0.0	J

ZEROES	REAL	IMAGINARY	
1	2.2033545	0.0	J

GAIN CONSTANT = 83.810272

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	23.852104
2	35.853822
3	13.522222

B VECTOR

3	83.810272

C VECTOR

1	2.2033545
2	1.0000000

REMARKS:

ASK FOR THREE POLES ONE ZERO.

RESULTS INDICATE TWO POLES WHEN THERE ARE THREE.

POLE AT -2.2 IS CANCELLED BY ZERO AT -2.22.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (f)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0934113	0.0	J
2	-10.207312	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 90.3204350

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	11.232745
2	11.300723

B VECTOR

2	90.320435

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

RESULTS INDICATE TWO POLES NO ZEROES WHEN THERE ARE THREE. METHOD NEGLECTED POLE AT -100.0, OTHERWISE IDENTIFICATION VERY GOOD.

EXAMPLE 3

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-7.8	-51.5	J
2	-7.8	51.5	J
ZEROES	REAL	IMAGINARY	

GAIN CONSTANT = 2950.0

TESTS FOR EXAMPLE 3

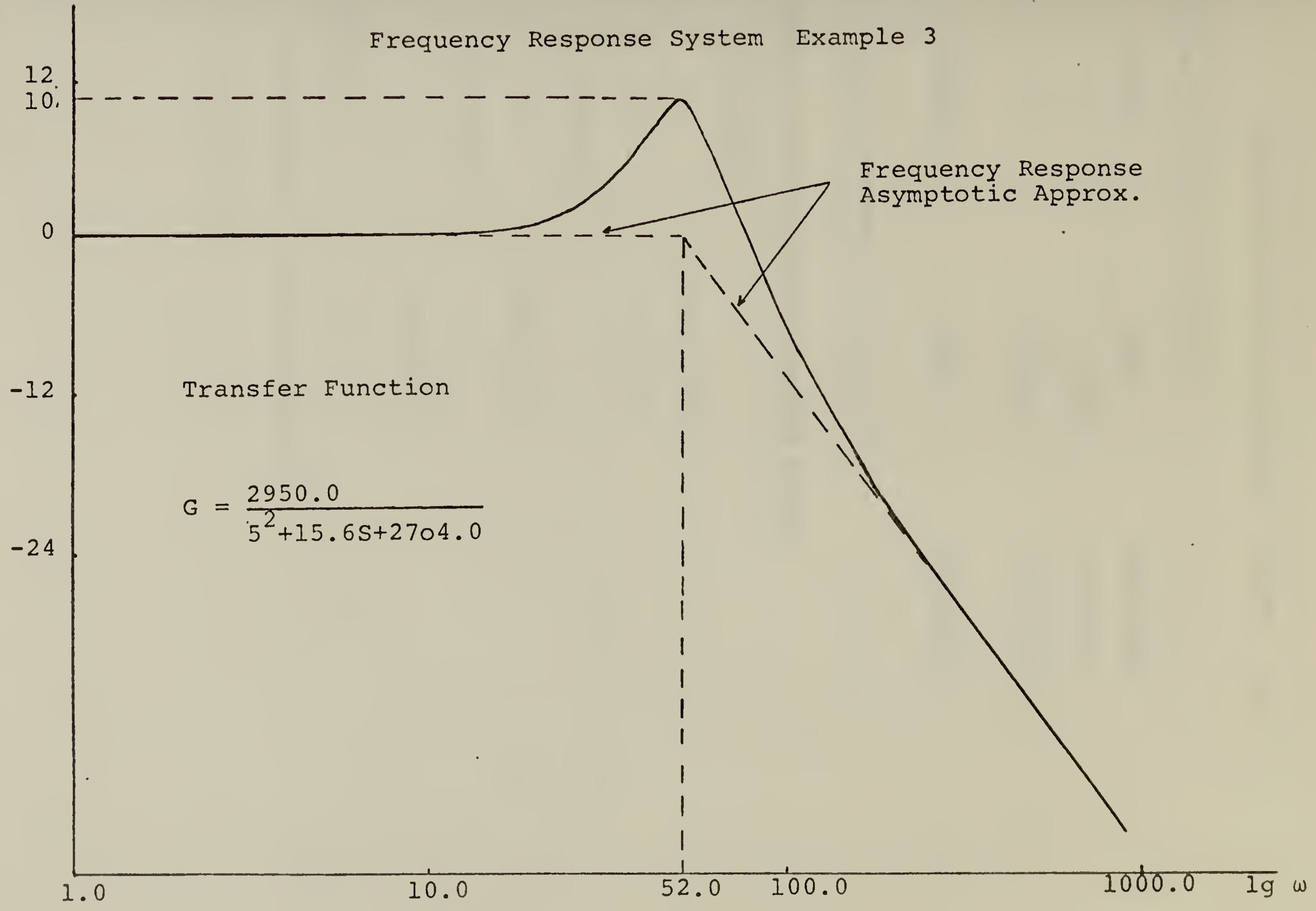
RUN	NO. OF POLES	NO. OF RESULTS	RESULTS
a	2	0	Good

D.C. POSITIONING SERVO FREQUENCY RESPONSE IN FIG. 2

INPUT FUNCTION STEP-WISE INCREASING SIMILAR TO THE ONE SHOWN IN FIG. 1, BUT WITH STEP AMPLITUDE ABOUT 0.5 VOLTS.

INITIAL CONDITIONS EQUAL ZERO.

FIGURE 2


Frequency Response System Example 3
Mdb
12
10
0
-12
-24
Frequency Response
Asymptotic Approx.
Transfer Function
G = 2950.0 / S^2+15.6S+27o4.0
1.0
10.0
52.0
100.0
1000.0
lg ω

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-5.3585933	-56.771298	J
2	-5.3585933	56.771298	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 3332.2095

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	3251.6948
2	10.717187

B VECTOR

2	3332.2095

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

GOOD RESULTS.

EXAMPLE 4

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-4.9	0.0	J
2	-32.0	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 150.0

TESTS FOR EXAMPLE 4

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	2	0	Good

D.C. POSITIONED SERVO.

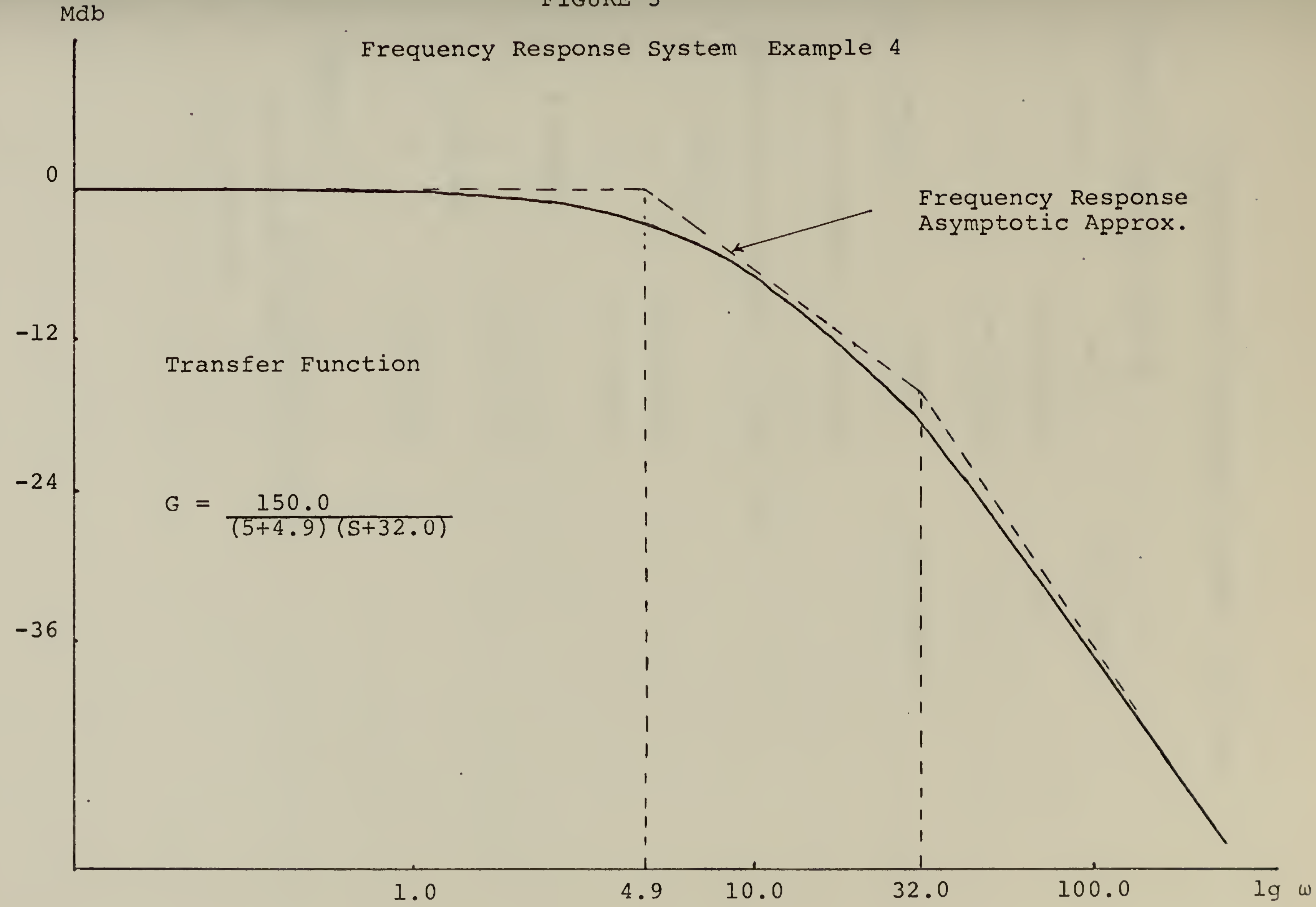
FREQUENCY RESPONSE IN FIG. 3.

INPUT FUNCTION THE SAME AS IN EXAMPLE 2.

INITIAL CONDITIONS EQUAL ZERO.

FIGURE 3

Frequency Response System Example 4



IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-5.2129680	0.0	J
2	-36.525846	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 178.37120

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	190.40807
2	41.738815

B VECTOR

2	178.37120

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

GOOD RESULTS.

EXAMPLE 5

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0	0.0	J
2	-10.0	0.0	J
3	-55.0	0.0	J

ZEROES	REAL	IMAGINARY	
1	-5.0	0.0	J

GAIN CONSTANT = 500.0

TESTS FOR EXAMPLE 5

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	3	1	Good
b	2	0	See Table of Data

INPUT FUNCTION STEP-WISE STRICTLY INCREASING SIMILAR TO THE ONE SHOWN IN FIG. 1.

INITIAL CONDITIONS EQUAL ZERO.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-0.89534766	0.0	J
2	-8.3914174	0.0	J
3	-88.600646	0.0	J

ZEROES	REAL	IMAGINARY	
1	-3.8670622	0.0	J

GAIN CONSTANT = 706.88770

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	665.67756
2	830.32662
3	97.887411

B VECTOR

3	706.88770

C VECTOR

1	3.8670622
2	1.0000000

REMARKS:

ASK FOR THREE POLES ONE ZERO.

RESULTS SHOW POOR IDENTIFICATION OF THE HIGHER FREQUENCY POLE.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-3.2441577	0.0	J
2	-108.41738	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 763.43213

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	351.72309
2	111.66154

B VECTOR

2	763.43213

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

RESULTS CAN BE CONSIDERED AS A LOWER ORDER MATHEMATICAL MODEL.

EXAMPLE 6

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.0	0.0	J
2	-10.0	0.0	J
3	-33.0	0.0	J
4	-100.0	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 660000

TESTS FOR EXAMPLE 6

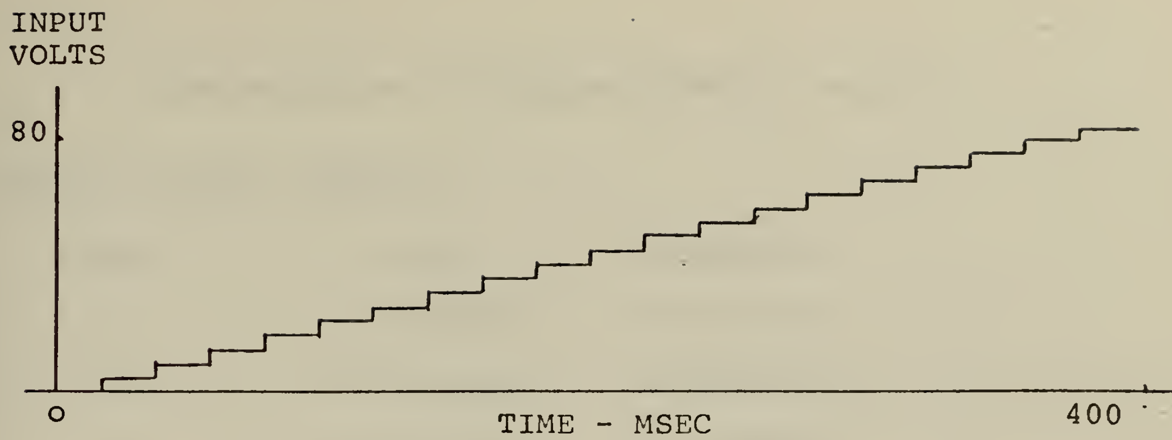
RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	4	0	Bad
b	4	0	Good
c	6	2	See Table of Data
d	4	2	See Table of Data
e	5	2	See Table of Data
f	4	1	See Table of Data
g	5	1	See Table of Data
h	3	0	See Table of Data

INPUT FUNCTION FOR RUN (a) STEP-WISE STRICTLY INCREASING SHOWN IN FIG. 4.

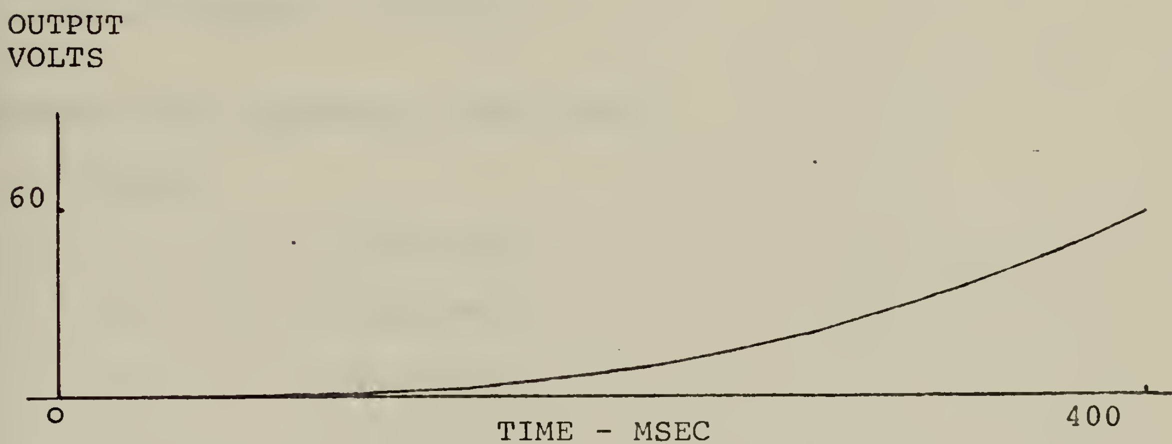
INPUT FUNCTION FOR RUNS (b), (c), (d), (e), (f), (g), and

(h) STEP-WISE INCREASING-DECREASING SHOWN IN FIG. 5.

INITIAL CONCLUSION EQUAL ZERO.



FIRST INPUT FUNCTION USED IN EXAMPLE NO. 6



OUTPUT CORRESPONDENT TO THE ABOVE INPUT

FIGURE 4

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.5013900	-4.3496191	J
2	-1.5013900	4.3496191	J
3	-5.4821332	-26.089080	J
4	-5.4321332	26.089080	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 62546.430

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	15047.775
2	2366.2077
3	764.79050
4	13.967046

B VECTOR

4	62546.430

C VECTOR

1	1.0000000

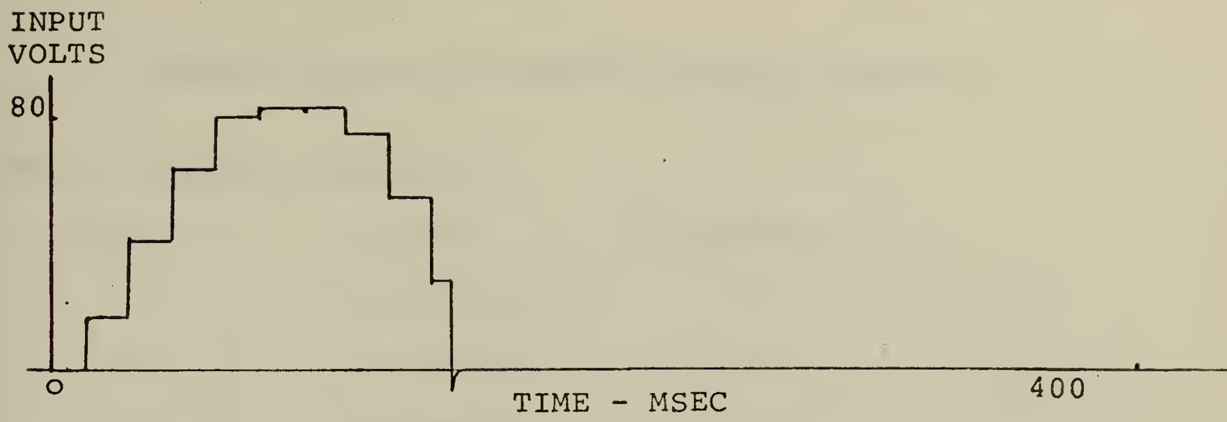
REMARKS:

INPUT FUNCTION - STEP-WISE INCREASING FUNCTION SHOWS IN FIG. 4.

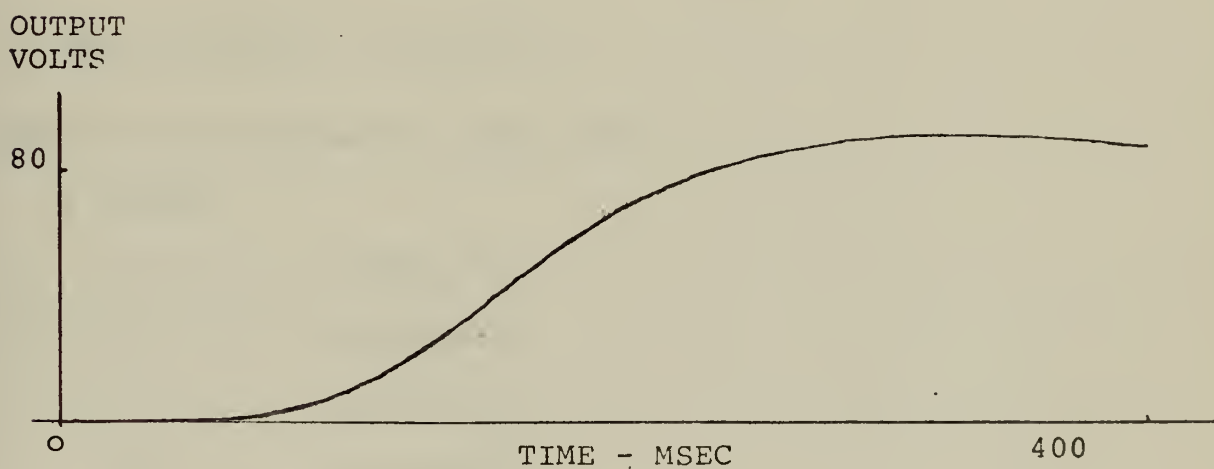
ASK FOR FOUR POLES NO ZEROES.

RESULTS NOT GOOD.

NOTE THE POSITIVE SECOND DERIVATIVE OF THE OUTPUT AT THE END OF THE SAMPLING INTERVAL.



SECOND INPUT FUNCTION USED IN EXAMPLE NO. 6



OUTPUT CORRESPONDENT TO THE ABOVE INPUT

FIGURE 5

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.0172066	0.0	J
2	-10.764637	0.0	J
3	-33.596973	0.0	J
4	-105.65671	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 612357.94

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	77080.941
2	48396.117
3	5351.3789
4	152.03553

B VECTOR

4	612357.94

C VECTOR

1	1.0000000

REMARKS:

INPUT FUNCTION STEP-WISE INCREASING-DECREASING SHOWN IN FIG. 5.

ASK FOR FOUR POLES NO ZEROES.

RESULTS VERY GOOD.

NOTE THE NEGATIVE SECOND DERIVATIVE DURING SOME TIME WITHIN THE SAMPLING INTERVAL.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.0375228	0.0	J
2	-10.649015	0.0	J
3	-34.253558	0.0	J
4	-101.99791	0.0	J
5	6.7412854	-96.372163	J
6	6.7412854	96.372163	J

ZEROES	REAL	IMAGINARY	
1	6.2635645	-96.069197	J
2	6.2635645	-96.069197	J

GAIN CONSTANT = 603327.00

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	0.7075898D 09
2	0.44024818D 09
3	48381252.0
4	1366621.4
5	12569.020
6	135.45544

B VECTOR

6	603327.00

C VECTOR

1	9268.5229
2	-12.527129
3	1.0000000

REMARKS:

ASK SIX POLES TWO ZEROES.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE COMPLEX POLES ARE CANCELLED BY COMPLEX ERRORS.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (d)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.0305074	0.0	J
2	-10.677930	0.0	J
3	-34.308456	0.0	J
4	-96.072665	0.0	J

ZEROES	REAL	IMAGINARY	
1	-516.60587	0.0	J
2	751.83224	0.0	J

GAIN CONSTANT = -1.4570827

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	71464.878
2	44715.215
3	4974.7267
4	143.08956

B VECTOR

4	-1.4570827

C VECTOR

1	-388400.95
2	-235.22636
3	1.0000000

REMARKS:

ASK FOR FOUR POLES TWO ZEROES.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE THE TWO EXCESS ZEROES APPEAR ONE IN THE R.H.P. AND THE OTHER IN THE L.H.P. BUT CAN BE NEGLECTED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (e)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.7344050	0.0	J
2	-8.8026617	0.0	J
3	-43.843968	0.0	J
4	-68.587703	0.0	J
5	8.0082667	0.0	J

ZEROES	REAL	IMAGINARY	
1	7.9267041	0.0	J
2	-622.31546	0.0	J

GAIN CONSTANT = 777.31934

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	-579657.55
2	-227126.83
3	2737.3550
4	3335.5841
5	115.96047

B VECTOR

5	777.31934

REMARKS:

ASK FOR FIVE POLES TWO ZEROES

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE THE EXCESS POLE AT 8.0 IS CANCELLED BY THE ZERO AT +7.9 AND THE OTHER EXCESS ZERO AT - 622.3 CAN BE NEGLECTED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (f)

SYSTEM TRANSFER FUNCTIONS

POLES	REAL	IMAGINARY	
1	-2.0267057	0.0	J
2	-10.704298	0.0	J
3	-34.037221	0.0	J
4	-101.07630	0.0	J

ZEROES	REAL	IMAGINARY	
1	-4045.2488	0.0	J

GAIN CONSTANT = 146.26987

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	74636.682
2	46730.407
3	5182.1817
4	147.84453

B VECTOR

4	146.26987

C VECTOR

1	4045.2488
2	1.0000000

REMARKS:

ASK FOR FOUR POLES ONE ZERO

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE

SYSTEM, SINCE THE EXCESS ZERO ZT - 4045.2 CAN BE NEGLECTED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (g)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.1810404	0.0	J
2	-10.145443	0.0	J
3	-35.910038	0.0	J
4	-96.915087	0.0	J
5	16.571798	0.0	J

ZEROES	REAL	IMAGINARY	
1	16.789359	0.0	J

GAIN CONSTANT = 580130.69

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	-1276179.1
2	-682609.54
3	-39334.691
4	2734.1957
5	128.57981

B VECTOR

5	580130.69

C VECTOR

1	-16.789359
2	1.0000000

REMARKS:

ASK FOR FIVE POLES ONE ZERO.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE EXCESS POLE AT 16.5 IS CANCELLED BY EXCESS ZERO AT 16.7.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (h)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0271020	0.0	J
2	-15.051165	-9.0280041	J
3	-15.051165	9.0280041	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 4051.1528

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	316.39101
2	338.96060
3	31.129433

B VECTOR

3	4051.1528

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

RESULTS CAN BE CONSIDERED AS A LOWER ORDER MATHEMATICAL MODEL.

EXAMPLE 7

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-25.0	0.0	J
2	-67.0	0.0	J
3	-150	-185.3	J
4	-150	185.3	J

ZEROES	REAL	IMAGINARY	
1	-3.1	0.0	J
2	-10.0	0.0	J

TESTS FOR EXAMPLE 7

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	6	3	See Table of Data
b	4	2	Good
c	4	3	See Table of Data

INPUT FUNCTION SIMILAR TO THE ONE SHOWN IN FIG. 5.

INITIAL CONDITIONS EQUAL TO ZERO.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-26.773050	0.0	J
2	-72.749634	0.0	J
3	-101.14084	173.81324	J
4	-101.14084	-173.81324	J
5	960.71376	0.0	J
6	52.080470	0.0	J

ZEROES	REAL	IMAGINARY	
1	50.693422	0.0	J
2	-3.0909162	0.0	J
3	-9.5071748	0.0	J

GAIN CONSTANT = -0.33196749E 09

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	0.39410705D 13
2	0.14131408D 12
3	-0.12683604D 10
4	-43800405.
5	-193111.44
6	-710.98986

B VECTOR

6	-0.33196749E 09

C VECTOR

1	-1489.6708
2	-609.25446
3	-38.095331
4	1.0000000

REMARKS:

ASK FOR SIX POLES THREE ZEROES.

RESULTS INDICATE FOUR POLES TWO ZEROES, SINCE POLE AT 52.0 IS CANCELLED BY ZERO AT 50.7 AND POLE AT 960.7 CAN BE NEGLECTED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-26.992965	0.0	J
2	-77.607958	0.0	J
3	-126.52085	-182.98579	J
4	-126.52085	182.98579	J

ZEROES	REAL	IMAGINARY	
1	-3.1577742		
2	-9.376548.	0.0	J

GAIN CONSTANT = 444891.63

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	0.10367784D 09
2	5706927.6
3	78054.592
4	357.64263

B VECTOR

4	444891.63

C VECTOR

1	29.609022
2	12.534322
3	1.0000000

REMARKS:

ASK FOR FOUR POLES TWO ZEROES.

GOOD RESULTS.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-27.385227	0.0	J
2	-75.010363	0.0	J
3	-119.37880	-181.29378	J
4	-119.37880	181.29378	J

ZEROES	REAL	IMAGINARY	
1	-3.1367869	0.0	J
2	-9.4693811	0.0	J
3	-5036.4256	0.0	J

GAIN CONSTANT = 82.045258

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	96790162.0
2	5315200.5
3	73620.633
4	341.15318

B VECTOR

4	82.045258

C VECTOR

1	149599.12
2	63519.731
3	5049.0317

4 1.0000000

REMARKS:

ASK FOR FOUR POLES THREE ZEROES.

RESULTS GIVE CLEAR INDICATION OF THE ORDER OF THE SYSTEM, SINCE ZERO AT -5036.4 CAN BE NEGLECTED.

EXAMPLE 8

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0	0.0	J
2	-10.0	0.0	J
3	25.0	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2500

TESTS FOR EXAMPLE 8

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	3	0	Good
b	3	0	Good
c	3	0	Good

INPUT FUNCTION SIMILAR TO THE ONE SHOWN IN FIG. 5.

INITIAL CONDITIONS DIFFERENT FROM ZERO FOR THE THREE DIFFERENT RUNS.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-0.97837856	0.0	J
2	-10.976897	0.0	J
3	-24.694433	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2378.9880

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	265.20737
2	305.93833
3	36.649709

B VECTOR

3	2378.9880

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

GOOD RESULTS WITH I.C. DIFFERENT FROM ZERO.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-0.96861216	0.0	J
2	-10.956655	0.0	J
3	-24.782535	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2379.6726

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	263.01082
2	306.15108
3	36.707801

B VECTOR

3	2379.6726

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

GOOD RESULTS WITH I.C. DIFFERENT FROM ZERO.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-24.839515	0.0	J
2	-0.95989923	0.0	J
3	-10.973508	0.0	J

ZEROES REAL IMAGINARY

GAIN CONSTANT = 2386.2974

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	261.64609
2	306.95352
3	36.772923

B VECTOR

3	2386.2974

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

GOOD RESULTS WITH I.C. DIFFERENT FROM ZERO.

EXAMPLE 9

SYSTEM TO BE IDENTIFIED

SAME SYSTEM EXAMPLE 8.

TESTS FOR EXAMPLE 9

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	3	0	Bad
b	3	0	Almost Good
c	3	0	Good
d	3	0	Good

INPUT FUNCTION STEP-WISE INCREASING-DECREASING SIMILAR TO THE ONE SHOWN IN FIG. 5.

INITIAL CONDITIONS EQUAL ZERO.

LOW FREQUENCY NOISE - FROM A FEW CYCLES TO ABOUT 300. CYCLES PER SECOND - WAS ADDED AT THE INPUT AND OUTPUT.

DIFFERENT R.M.S. VALUES OF NOISE WERE CONSIDERED.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-0.25785674D-01	0.0	J
2	-16.598045	-7.2418627	J
3	-16.598045	7.2418627	J

ZEROES REAL IMAGINARY

GAIN CONSTANT = 2294.0054

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	8.4561459
2	328.79567
3	33.221877

B VECTOR

3	2294.0054

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

R.M.S. VALUE OF THE NOISE 1.0 VOLTS.

RESULTS NOT GOOD.

IDENTIFICATION OF UNKNOWN SYSTEM - RUN (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-23.206882	0.0	J
2	-12.305073	0.0	J
3	-0.72464570	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2364.4692

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	206.93154
2	311.29595
3	36.236600

B VECTOR

3	2364.4692

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

R.M.S. VALUE OF THE NOISE .75 VOLTS.

RESULTS ALMOST GOOD.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.1475703	0.0	J
2	-10.808421	0.0	J
3	-23.475558	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2289.9822

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	291.17727
2	293.07699
3	35.431549

B VECTOR

3	2289.9822

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

R.M.S. VALUE OF THE NOISE 0.5 VOLTS.

RESULTS GOOD.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (d)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0495050	0.0	J
2	-10.680594	0.0	J
3	-25.023091	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 2381.8215

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	280.49226
2	304.73267
3	36.753190

B VECTOR

3	2381.8215

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

R.M.S. VALUE OF THE NOISE .25 VOLTS.

RESULTS VERY GOOD.

EXAMPLE 10

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0	0.0	J
2	-10.0	0.0	J
3	-20.0	0.0	J
4	-33.0	0.0	J
5	-100.0	0.0	J

ZEROES	REAL	IMAGINARY	
1	-5.0	0.0	J

GAIN CONSTANT = 2640000.0

TESTS FOR EXAMPLE 10

RUN	NO. OF POLES	NO. OF ZEROES	RESULT
a	5	1	See Table of Data
b	3	0	See Table of Data
c	2	0	See Table of Data

INPUT FUNCTION SIMILAR TO THE ONE SHOWN IN FIG. 5.

INITIAL CONDITIONS EQUAL ZERO.

SEE FIG. 6 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-0.94505946	0.0	J
2	-10.807637	0.0	J
3	-32.034223	0.0	J
4	-62.693979	0.0	J
5	-120.97375	0.0	J

ZEROES	REAL	IMAGINARY	
1	-4.9340575	0.0	J

GAIN CONSTANT = 2741280.0

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	248.538.7
2	2992971.5
3	403446.21
4	16013.272
5	227.45465

B VECTOR

5	2741280.0

C VECTOR

1	4.9340575
2	1.0000000

REMARKS:

ASK FOR FIVE POLES ONE ZERO.

RESULTS NOT VERY GOOD FOR THE HIGHER FREQUENCY POLES.

SEE FIG. 6 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-2.6408186	0.0	J
2	-24.059549	0.0	J
3	-24.059549	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 10410.082

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	3331.3020
2	1388.5395
3	50.759916

B VECTOR

3	10410.082

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

RESULTS CAN BE CONSIDERED A LOWER ORDER MATHEMATICAL MODEL.

SEE FIG. 6 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-5.6608772	0.0	J
2	-13.680660	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 182.89790

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	77.444535
2	19.341537

B VECTOR

2	182.89790

C VECTOR

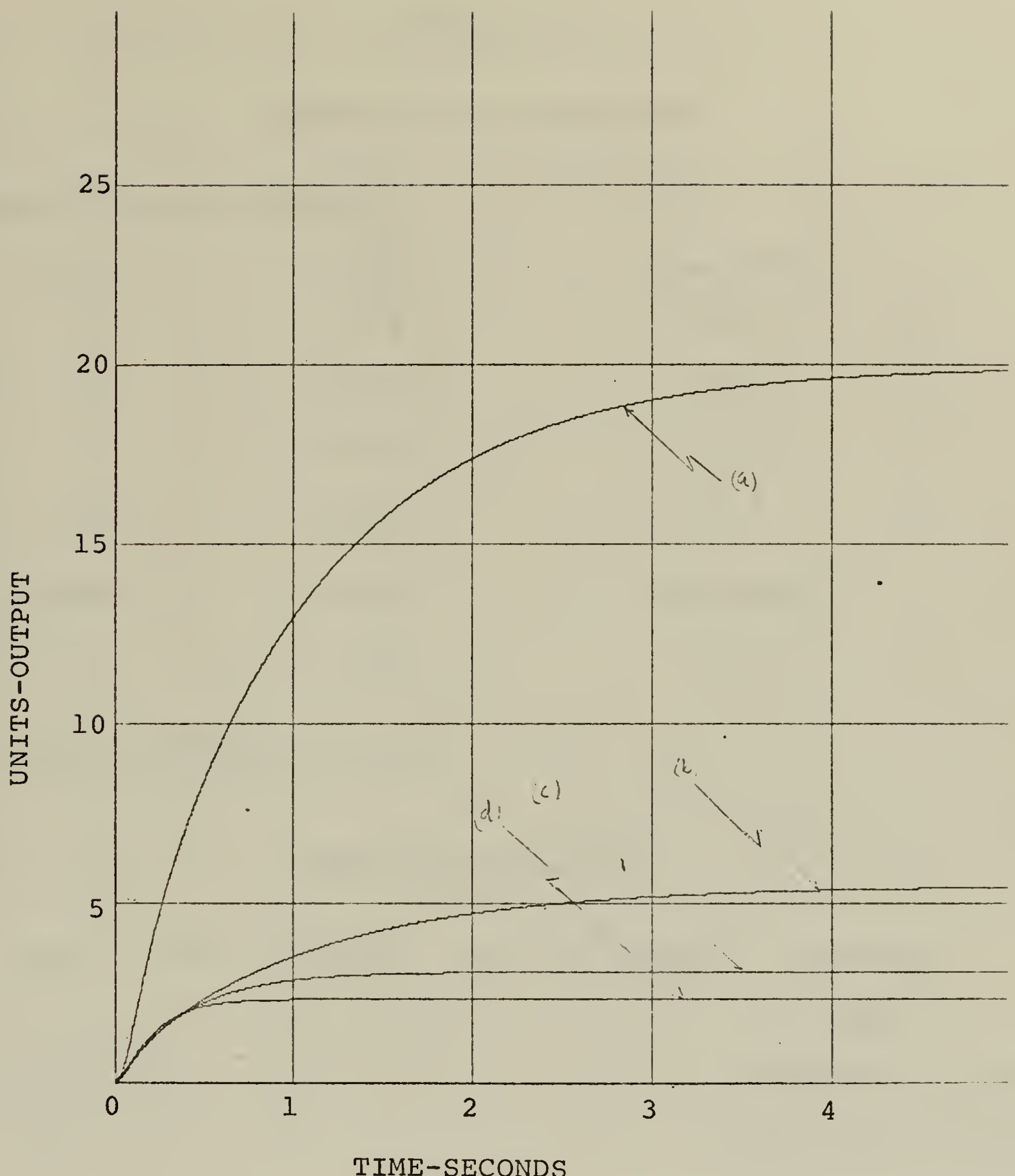
1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

RESULTS CAN BE CONSIDERED A LOWER ORDER MATHEMATICAL MODEL.

SEE FIG. 6 COMPARING RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.



UNIT STEP RESPONSE OF:

(a) System to be identified
(b) Fifth order identified model
(c) Third order identified model
(d) Second order identified model

FIGURE 6

EXAMPLE 11

SYSTEM TO BE IDENTIFIED

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0	0.0	J
2	-10.0	0.0	J
3	-33.0	0.0	J
4	-100.0	0.0	J

ZEROES	REAL	IMAGINARY	
1	-5.0	0.0	J

GAIN CONSTANT = 52500.0

TESTS FOR EXAMPLE 11

RUN	NO. OF POLES	NO. OF ZEROES	RESULTS
a	4	1	Good
b	3	0	See Table of Data
c	2	0	See Table of Data

INPUT FUNCTION IS SIMILAR TO THE ONE SHOWN IN FIG. 5.

INITIAL CONDITIONS EQUAL ZERO.

SEE FIG. 7 & 8 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFYING SYSTEMS TO A STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (a)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-1.0182594	0.0	J
2	-10.853145	0.0	J
3	-35.841486	0.0	J
4	-108.22539	0.0	J

ZEROES	REAL	IMAGINARY	
1	-5.0491588	0.0	J

GAIN CONSTANT = 64711.781

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	42867.599
2	47640.814
3	5600.2860
4	155.93828

B VECTOR

4	64711.781

C VECTOR

1	5.0491588
2	1.0000000

REMARKS:

ASK FOR FOUR POLES ONE ZERO.

RESULTS GOOD.

SEE FIG. 7 & 8 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (b)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-31.063053	-36.791868	J
2	-31.063053	36.791868	J
3	-2.9265019	0.0	J

ZEROES REAL IMAGINARY

GAIN CONSTANT = 28337.477

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	6785.2550
2	2500.3670
3	65.052607

B VECTOR

3	28337.477

C VECTOR

1	1.0000000

REMARKS:

ASK FOR THREE POLES NO ZEROES.

RESULTS CAN BE CONSIDERED A LOWER ORDER MATHEMATICAL MODEL.

SEE FIG. 7 & 8 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

IDENTIFICATION OF UNKNOWN SYSTEM - Run (c)

SYSTEM TRANSFER FUNCTION

POLES	REAL	IMAGINARY	
1	-3.4314173	0.0	J
2	-36.703534	0.0	J

ZEROES	REAL	IMAGINARY

GAIN CONSTANT = 537.30396

SYSTEM STATE VARIABLES (PHASE FORM)

A VECTOR

1	125.94514
2	40.134951

B VECTOR

2	537.30396

C VECTOR

1	1.0000000

REMARKS:

ASK FOR TWO POLES NO ZEROES.

RESULTS CAN BE CONSIDERED A LOWER ORDER MATHEMATICAL MODEL.

SEE FIGs. 7 & 8 COMPARING THE RESPONSES OF THEORETICAL AND IDENTIFIED SYSTEMS TO A UNIT STEP INPUT.

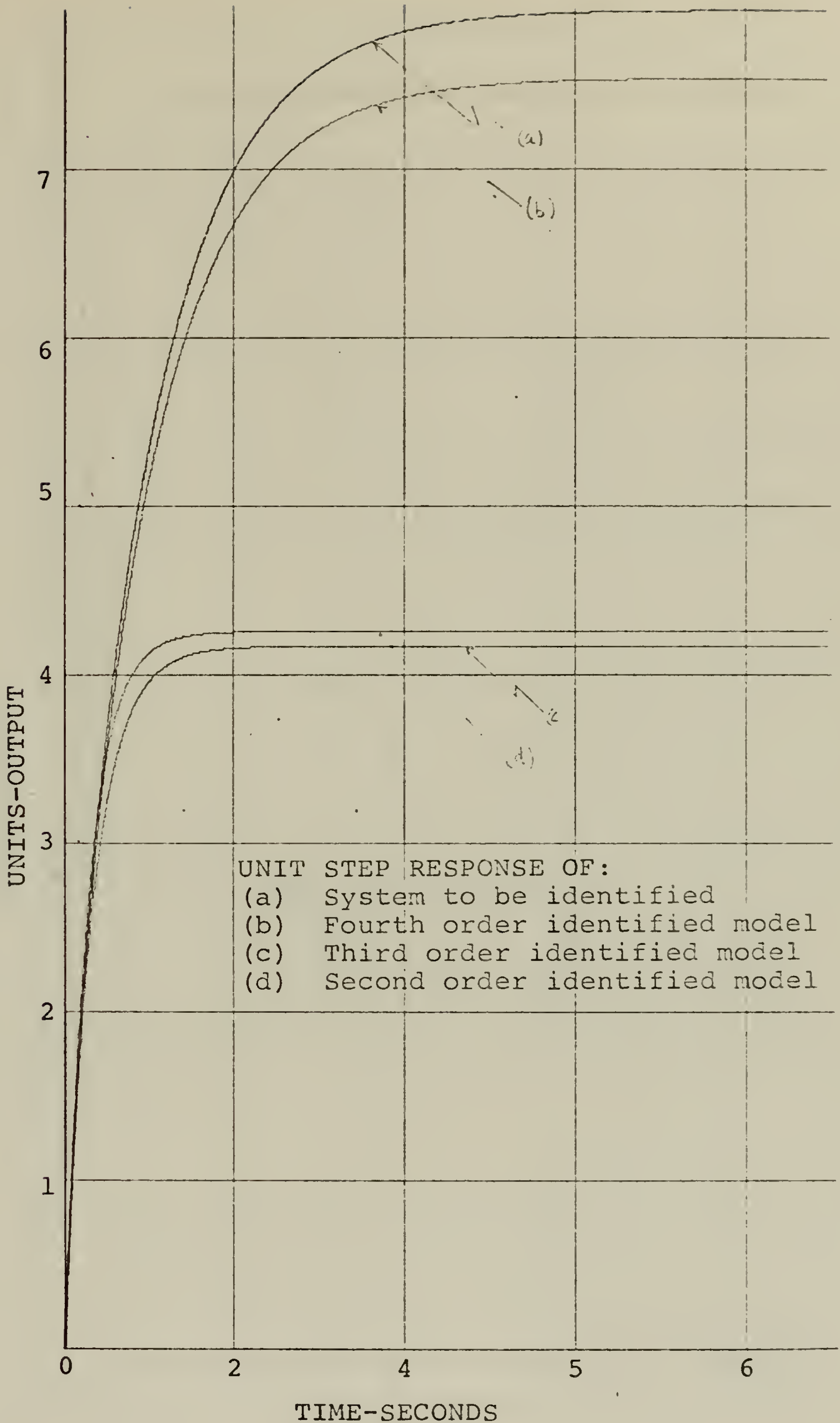
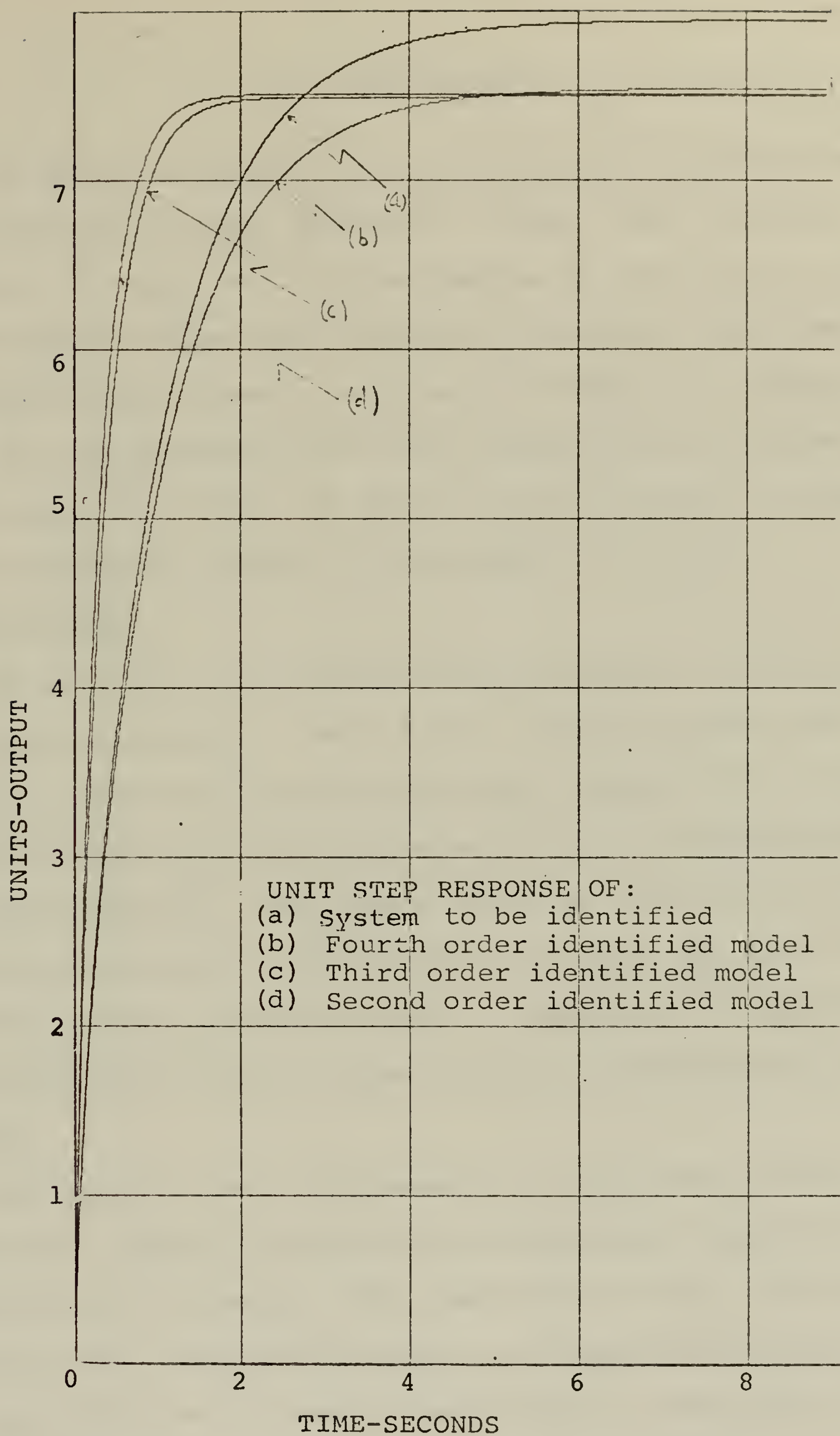

UNITS-OUTPUT
1
2
3
4
5
6
7
0
2
4
5
6
TIME-SECONDS
(a)
(b)
UNIT STEP RESPONSE OF:
(a) System to be identified
(b) Fourth order identified model
(c) Third order identified model
(d) Second order identified model

FIGURE 7



UNIT STEP RESPONSE OF:
(a) System to be identified
(b) Fourth order identified model
(c) Third order identified model
(d) Second order identified model
(a)
(b)
(c)
(d)
UNITS-OUTPUT
TIME-SECONDS
0
2
4
6
8
1
2
3
4
5
6
7

FIGURE 8

IV. CONCLUSIONS

The method of multiple integrations is a practical method for identifying lumped parameter, linear, time invariant systems. It works well in the absence of noise up to a fourth order system if the input function is carefully selected. For higher order systems the identification is in general not good, but an approximation of the system can be obtained. If desired, the method can also be used to obtain a reduced order mathematical model of the system, after the identification.

The accuracy of an identification depends very strongly on the noise present. Small R.M.S. values of noise are enough to make the identification bad. Noise with a R.M.S. value of about 0.5 volts and below do not practically affect the identification, when inputs like the ones ued in this research are present, independently of where the noise is present - input, output or both -. Accuracy also depends very much on the type of input function used to drive the system.

The computational requirements of the method are not excessive, however, an accurate A/D converter system is very desirable and the better this operation is done, the more accurate will the identification be, since the accuracy of an identification is usually comparable to the number of exact significant digits in the input-output data.

The limitation of the method to low order systems is primarily due to the algorithm used to solve the over-determined set of linear equations. As better algorithms become available, it will be possible to identify higher order systems.

The results obtained when reduction of the system order was desired were not very good at first sight, but when the steady state gain of the lower order models was adjusted, making it equal to the one of the non-reduced identified model, the approximation can be considered good. However, when a pair of dominant complex roots is present in the system a better reduced order model should be expected.

APPENDIX I

A. OBTAINING THE INPUT - OUTPUT RECORD

The identification technique described in Chapter II is based on knowledge of the input and output record and since the IBM 360 Digital Computer was used to perform the integrations of equation (5), the record should be available in a digital form. Thus, and A/D (analog to digital) operation should be performed. This A/D operation was accomplished using the Hybrid Computer System (XDS 9300/Ci 5000) located in the Electrical Engineering Computer Laboratory. The output of this computer is a seven track magnetic tape in which the input and output functions are recorded in digital form. Since the XDS 9300/Ci 5000 system is used on a self-service basis, a brief discussion of what was done to obtain the record, is given here, so that the same procedure could be used in the future whenever an A/D conversion is needed - it should be pointed out here than the computer center staff was always predisposed to assist in the operation of the system.

The analog part of the system, the Ci 5000, has two patch-boards which are the heart of the system. The logic patch-board was utilized to set the sampling rate, at which the analog signal was sampled - 10KHZ in this particular case.

This sampling rate is the maximum that can be achieved with this computer, and can be changed (decreased) by altering a counter-like number display located in the

frontal panel of the computer - FP00 and FP04. Also in this patchboard the base frequency - 1KHZ - used to determine the moments at which the input function is changed step-wise, is set. The same counter-like number display can be used to change this frequency - FF10 and 14. In the digital patch-board, the wiring that controls the digital part of the A/D operation is also set.

The analog patchboard was used to introduce the proper isolation between the different components of the different systems tested and to provide the desired amplification factor. This patchboard was also used in the generation of the input signal. This signal was obtained in the following way. An oscillator generates a sine wave in which amplitude and frequency can be adjusted by two potentiometers - P043 and P417 -; the sine-wave is amplified and passed through a zero order hold circuit which output is a step-wise wave form, that was used as the input function. The sine wave was also used to control the beginning of the sampling. The wiring of both patchboards was set by the computer center staff and those patchboards - numbered 21 and 21 - are reserved for the A/D operation.

The input and output signals are fed from the analog part of the computer system to the digital part. The Digital part interfaced with the analog is a Xerox Data System - XDS9300.

Two tapes drive units are available as input-output devices. One printer is used for program listings and

21. With the sense switches at the XDS9300 console off, depress CLEAR and CLEAR FLAGS buttons simultaneously.

22. Mount the magnetic tape following the instructions available on the inside of the protective doors.

23. Select density 556 bits per inch with the corresponding knob on the face of the tape driver.

24. Read in the A/D program, described in Chapter III and listed in Chapter IV, following the instructions available on the console of the XDS9300.

25. Enter, through the teletype, the parameters to be used for the A/D program. This program has seven options or instructions which are entered also through the teletype by a single digit number and depressing the carriage return button. If a further response is required the computer would indicate the format. The options are:

1 = Desire to enter new parameters.

2 = Start digitizing the analog input signals (normally actuated by manual switch operation on Ci 5000 control panel; (DS1 up and key COMPUTE depressed).

3 = Write an END OF FILE on the magnetic tape.

4 = Rewind the tape to the load point (beginning).

5 = Skip the following number of END OF FILE marks.

6 = Print the following amount of digitized data.

7 = Actuate the A/D program for the next single block of data encountered and feed the signal to the strip chart recorder (which must be running prior to this option selection connected to the analog patchboard on the Ci 5000.

The rest of the counters should be set to zero.

7. Depress the master clock control, RUN.

8. Set DF00 (Delay flip-flop) to .1 ms. and adjust it, until its corresponding light on the front panel of the Ci 5000 comes on.

9. Set DF06 in .1 ms., DF07 in 1 sec. and DF10 in 1 ms.

10. Set all the DS (digital switches) to the center position.

11. Turn on the TRC.

12. Set switches X1 to the left and X2 to the right.

13. Select amplifier 042 in the Ci 5000 console, adjusting P417, DF06 and F10, F14 to obtain the desired input function which can be observed in the TRC when the DS1 switch is set in the up position and the button COMPUTE is depressed in the Ci 5000 console.

14. Set the R.T. clock switch on XDS9300 to EXIT.

15. If the XDS9300 is not energized, depress simultaneously RESET and POWER buttons.

16. If the teletype is not ON, energize it by turning the knurled knob to ON.

17. If hte Line Printer is not ON, depress the POWER switch - indicator on the face of the printer. About one minute later the lower half of the Power button/lights up indicating the printer is energized.

18. Depress READY SWITCH INDICATOR.

19. Depress the POWER ON button on the card reader.

20. Depress IDLE button on the console of the XDS9300.

outputs. A card reader is associated with the digital computer as the avenue of input for compiling programs. A teletype unit is used for parameter inputs and instructions to the digital computer program. A control console displays information of the computer and has the corresponding control switches. A recorder is also available with which the records of the input and output, after the A/D operation is finished, can be recorded in a strip chart.

After the preceding general description of the system, the following procedure can be followed to take a record.

1. Set the analog and digital patchboards in the Ci 5000.

2. Connect the output of the amplifier 042 to the input of the system.

3. Connect the output of the system to the input of the amplifier 036.

4. If the Ci 5000 is not energized, depress on switch indicator at lower left corner of display panel of Ci 5000. Depress KEY BOARD and POTSET buttons on the display panel.

5. Enter the maximum value of the sine wave through the POTSET, POT and SERVO keys.

6. Set the counters FF's in the following positions:

FF00 in 0, FF04 in 9, for 10KHZ sampling rate.

FF10 in 2, F14 in 4, for 40HZ rate of changing, step-wise, the input function.

FF30 in 0, if only one record per run is desired; increasing FF30 the number of records/run can be adjusted.

The parameters to be entered in point 25 are:

NSAMP = Total number of samples to be taken from each channel - 4002 in this case.

NCHAN = Number of channels to be digitized - 2 in this case.

NREC = Number of records to be digitized per run; this number should be equal to the one set in the counter FF30 (Ci 5000) plus 1; in this particular problem NREC = 1.

ITAPE = The identifying number of the tape drive in use. This number was set when the tape was mounted.

NDEL = Controls the delay between the digital and analog signal in option 7.

After entering the parameters the * character should be typed on the teletype followed by RETURN. At this point everything is ready to take a record; by typing, 2 - RETURN, on the teletype, and setting switch DS1 - UP and pressing COMPUTE on the Ci 5000. After the record was taken, write - 3 - (END OF FILE). If it is desired, the record can be recorded in a strip chart recorder.

B. SEVEN TO NINE TRACK TAPE CONVERSION

In the magnetic tape generated in point A, the input-output record is written in a seven track tape, but since this tape is going to be used with the IBM 360 Digital Computer and this computer uses NINE TRACK TAPES, a conversion of the seven track tape into a nine track tape is required.

This operation is done by the IBM 360, using the program CONVERT, which is described in the Technical Note No. 0211-08 second edition - PROCEDURES FOR CONVERTING 7-TRACK MAGNETIC TAPES TO 9-TRACK MAGNETIC TAPES by Sharon D. Raney [21]. This description is available at the Computer Center in Ingersol Hall. A listing of the CONVERT program is given in Chapter IV.

A/D PROGRAM

THIS PROGRAM IS TO BE USED WITH THE CI5000-SDS9300 HYBRID COMPUTER

```
      DIMENSION IBUF(6144,2),LOCB(-1:1),BADREC(100)
      INTEGER RECNUM,BADREC
      NAMELIST NREC, NSAMP, NCHAN, ITAPE,NDEL
 1    INPUT(101)
      NWORDS=NSAMP*NCHAN
      LOCB(-1)=LOCF(IBUF(1,1))
      IF NREC.GT.1,LOCB(1)=LOCF(IBUF(1,2))
      IF NREC.EQ.1,LOCB(1)=LOCB(-1)+NWORDS
      IND=2
      GO TO 15
 2    NB=1
      RECNUM=0
      NEWBUF=LOCB(1)
      IF IND.NE.2,GO TO 94
      CALL WRITECLOCK(0)
      CALL ADSTART(NCHAN,LOCB(-1),NEWBUF,NSAMP,RECNUM,11S)
      CALL MTRDY(ITAPE,LOCB(-1),LOCB(1),NWORDS,IND)
 4    IF IND.EQ.1, GO TO 4
      GO TO(4,3,91,93)IND
 3    IF(TEST(1).GT.0)GO TO 3
      NBAD=0
      CALL STARTCLOCK
      CALL ENABLE
 5    CONTINUE
 10   GO TO 5
 11   NB=-NB
```

```
      NEWBUF=LOCB(NB)
      IF(TEST(1).GT.0.OR.RECNUM.GE.NREC)CALL DISABLE
      GO TO(90,12,91,92)IND
12    CONTINUE
      CALL MTOUT(NB)
      IF(TEST(1).LT.0.AND.RECNUM.LT.NREC)GO TO 5
      CALL STOPCLOCK
      CALL ADSTOP
X     CALL PROCESS(IBUF,NSAMP,NCHAN+1,2S)
      OUTPUT(101)RECNUM
      IF NBAD.NE.0,WRITE(101,106) (BADREC(I),I=1,NBAD)
106   FORMAT(# BAD RECORDS ARE#,(/,I6))
15    OUTPUT(101)'OPTION=(I1)'
      READ(101,100)NOPT
100   FORMAT(I1)
      GO TO(1,2,30,40,50,60,70)NOPT
30    ENDFILE(ITAPE)
      OUTPUT(101)'EOF'
      GO TO 15
40    REWIND(ITAPE)
      GO TO 15
50    OUTPUT(101)'SKIPFILES=(I4)'
      READ(101,101)NF
101   FORMAT(I4)
      DO 55 I=1,NF
51    CALL BUFFERIN(ITAPE,1,IBUF(1,1),1,IND)
52    IF(IND.LT.2)GO TO 52
      IF(IND.NE.3)GO TO 51
55    CONTINUE
      OUTPUT(101)NF
      GO TO 15
60    OUTPUT(101)'NUMWORDS TO LIST=(I4)'
      READ(101,101)NW
      WRITE(101,105)NW,NCHAN
105   FORMAT(' WRITE ' I4 ' WORDS, ' I2 ' AT A TIME')
```

```
      IND=1
      CALL BUFFERIN(ITAPE,1,IBUF(1,1),NWORDS,IND)
66    IF(IND.EQ.1)GO TO 66
62    GO TO(62,63,64,65)IND
63     WRITE(6,102)
102   FORMAT(1H1)
      DO 631 I=1,NW,NCHAN
      WRITE(6,104)(IBUF(J,1),J=I,I+NCHAN-1)
104    FORMAT(12O10)
631   CONTINUE
      GO TO 15
64     OUTPUT(101)'EOF READ'
      GO TO 15
65     OUTPUT(101)'READ ERR'
      GO TO 63
70    OUTPUT(101)'START ANALOG RECORDER'
      OUTPUT(101)'TYPE * C/R TO CONTINUE'
      INPUT(101)
77    IND=1
      CALL BUFFERIN(ITAPE,1,IBUF,NWORDS,IND)
76    IF(IND.EQ.1)GO TO 76
71    GO TO(71,72,64,74)IND
72    DO 73 I=1,NWORDS
73    IBUF(I,1)=IBUF(I,1)/2**10
      DO 75 I=1,NWORDS,NCHAN
      DO 750 J=1,NCHAN
750   CALL DAC(J,IBUF(I+J-1,1))
      N=NDEL
      CALL DELAY
75    CONTINUE
      IF(SENSE SWITCH 1)77,15
74    OUTPUT(101)'READ ERROR'
      GO TO 72
90    CALL DISABLE
      CALL ADSTOP
```

```
      ØUTPUT(101)'RATE ERR',RECNUM
      GØ TØ 15
92    NBAD=NBAD+1
      BADREC(NBAD)=RECNUM-1
      GØ TØ 12
91    CALL DISABLE
      CALL ACSTØP
      ØUTPUT(101) ' MT NØT READY '
      GØ TØ 15
93    CALL DISABLE
      CALL ACSTØP
      ØUTPUT(101) ' MT ERRØR ØN SPACING FRØM LØAD PØINT'
      GØ TØ 15
94    ØUTPUT(101)'DELAY TIME BETWEEN RECØRDS TØØ SHØRT'
      GØ TØ 15
      END
```

```
      SUBRØUTINE PRØCESS(I,N,NC,IR)
      J=N*NC+9
      CALL DELAY
1     IF TEST(1).LT.0,GØ TØ 1
      IF TEST(2).GT.0,RETURN IR
      RETURN
      END
```

```
$ADSTART PZE
         BRM     9SETUPN
         PZE     6
NCH      PZE
BUF      PZE
NEWBUF   PZE
NSAM     PZE
RECNUM   PZE
NEXLOC   PZE
         LDA     ENDBRM
         XMA     040
         STA     SV040
         LDA     INTBRM
         XMA     052
         STA     SV052
         LDA     *NCH
         STA     NCH
         STA     INCR
         ADD     =COMM
         STA     COMLOC
         LDA     NCH
         LLSA    15
         ADD     =COMM
         STA     *COMLOC
         STA     CONTR
         LDA     *BUF
         STA     COMM
         STA     CLKPTR
         LDA     *NSAM
         SUB     =1
         STA     COUNT
         STA     NEWCT
         EOM     034001
         POT     CONTR
         BRR     ADSTART
```

```
INTBRM  BRM     ADFAST
ENDBRM  BRM     ENDAD
SVC40   PZE
ENDAD   PZE
        DIR
        HLT
        EOM     034001
        POT     CONTR
        EIR
        BRC     *ENDAD
$ADSTOP PZE
        LDA     SV040
        STA     040
        LDA     SV052
        STA     052
        STZ     *COMLOC
        MPB     ADSTOP
        BRR     ADSTOP
COMLOC  PZE
$ADFAST PZE
        DIR
        STD     SVAB
        LDA     INCR
        ADM     COMM
        LDA     R\CLOCK
        ETR     =01777
        ADM     *CLKPTR
        LDA     COMM
        STA     CLKPTR
        SKR     COUNT
        BRU     EXIT
        LDA     *NEWBUF
        STA     COMM
        STA     CLKPTR
        LDA     NEWCT
```

```
        STA      CØUNT
        MPØ     *RECNUM
        LDP      SVAB
        EIR
        BRC      *NEXLØC
EXIT    LDP      SVAB
        EIR
        BRC      *ADFAST
CØMM    RES      33
CØNTR   PZE
NEWCT PZE
SVAB    RES      2
INCR  PZE
CØUNT PZE
CLKPTR PZE
SV052   PZE
        END
```

```
A        EQU    5
B        EQU    4
X1       EQU    1
$MTRDY   PZE
         BRM    9SETUPN
         PZE    5
UNIT     PZE
BUF1     PZE
BUF2     PZE
NW       PZE
IR       PZE
I        EQU    IR
         LDB    *NW
         COPY   (0,A),(B,X1)
         LLSD   14
         MRG    ALCO
         STA    ALCN1
         STA    ALCN2
         COPY   (0,A)
         LDB    *BUF1
         LLSD   10
         LLSA   5
         ADM    ALCN1
         COPY   (X1,A)
         LLSD   14
         STA    BUTCW1
         COPY   (0,A)
         LDB    *BUF2
         LLSD   10
         LLSA   5
         ADM    ALCN2
         COPY   (X1,A)
         LLSD   14
         STA    BUTCW2
         LDA    =1
```

```
         STA     *IR
         LDB     *UNIT
         LDA     TRTO
         COPY    7,(B,A)
         STA     TRTN
         LDA     FPTO
         COPY    7,(B,A)
         STA     FPTN
         LDA     BTTO
         COPY    7,(B,A)
         STA     BTTN
         LDA     WTBO
         COPY    7,(B,A)
         STA     WTBN
         LDA     EFTO
         COPY    7,(B,A)
         STA     EFTN
         BRM     RDY
         BRU     RDYERR
         EXU     FPTN
         BRU     RDYERR
         EXU     BTTN
         BRM     BTSP
         MPO     *IR
         BRR     MTRDY.
*
RDYERR   MPT     *IR
         BRR     MTRDY
*
RDY      PZE
         EXU     TRTN
         CAT     0
         BPR     RDY
         MPO     RDY
         BRR     RDY
```

```
*
BTSP    PZE
        LDA    EØRBRM
        XMA    011
        STA    SV11
        EXU    EFTN
        EXU    ALCØ
        PØT    BTGAP
        MPØ    BTSP
        BRR    BTSP
IØC     FØRM   10,14
BTGAP   IØC    417,0
*
MTØUT   PZE
        BRM    ØSETUPN
        PZE    1
NB      PZE
        LDA    =1
        STA    *I
        BRM    RDY
        BRU    ERR
        LDA    EØRBRM
        XMA    011
        STA    SV11
        SKN    *NB
        BRU    ØUT2
        EXU    WTBN
        EXU    ALCN1
        PØT    ØUTCW1
        BRR    MTØUT
ØUT2    EXU    WTBN
        EXU    ALCN2
        PØT    ØUTCW2
        BRR    MTØUT
ERR     MPT    *I
```

```
         BRR     MT0JT
*
E0R      PZE
         BRM     RDY
         BRU     $+2
         CET     0
         MPT     *I
         MP0     *I
         XMA     SV11
         STA     011
         XMA     SV11
         BRC     *F0R
F0RBRM   BRM     E0R
SV11     PZE
*
TRTO     TRT     0,0
TRTN     PZE
FPTO     FPT     0,0
FPTN     PZE
BTTO     BTT     0,0
BTTN     PZE
WTBO     WTB     *0,0,4
WTBN     PZE
EFTO     EFT     *0,0,4
EFTN     PZE
ALCO     EOM     016000
ALCN1    PZE
ALCN2    PZE
OUTCN1   PZE
OUTCN2   PZE
         END
```

PROGRAM NO. 2

CONVERT PROGRAM

THIS PROGRAM CONVERTS A SEVEN TRACK TAPE TO A NINE TRACK TAPE

```
//CONVERT  EXEC  FORTCLG,REGION.GO=245K
//FORT.SYSIN DD *
      DIMENSION IDAT(8004),DAT(8004)
      FACTOR=100.0/(2**23)
      REWIND 2
      REWIND 4
      M=1
      LRECL=8004
   11 J=0
   10 READ(2,15,END=50,ERR=60) IDAT
   15 FORMAT(100(80A4),4A4)
      J=J+1
      WRITE (6,70) J
   70 FORMAT('0',10X,'RECORD NO.=',I4)
      CALL FORM(IDAT,LRECL)
      DO 22 I=1,LRECL
   22 DAT(I)=IDAT(I)*FACTOR
      WRITE (6,40) DAT
   40 FORMAT(1X,10F12.5)
      WRITE (4,15) DAT
      GO TO 10
   60 WRITE (6,61) J
   61 FORMAT('0',5X,'READ ERROR,RECORD NO.=',I3)
      GO TO 10
   50 WRITE (6,51) M,J
   51 FORMAT('0',5X,'END OF FILE ',I2,' RECORD NO.=',I3)
      END FILE 4
      M=M+1
      IF(M.LT.5) GO TO 11
      WRITE (6,71) M
   71 FORMAT('0',5X,'END OF TAPE,FILES=',I3)
      STOP
      END
//GO.FT06F001 DD  SPACE=(CYL,6)
//GO.FT02F001  DD  UNIT=2400-1,VOL=SER=MONTER,LABEL=(1,NL),
//      DISP=OLD,DCB=(DEN=1,RECFM=U,BLKSIZE=32016)
//GO.FT02F002  DD  UNIT=2400-1,VOL=SER=MONTER,LABEL=(2,NL),
//      DISP=OLD,DCB=(DEN=1,RECFM=U,BLKSIZE=32016)
//GO.FT02F003  DD  UNIT=2400-1,VOL=SER=MONTER,LABEL=(3,NL),
//      DISP=OLD,DCB=(DEN=1,RECFM=U,BLKSIZE=32016)
//GO.FT02F004  DD  UNIT=2400-1,VOL=SER=MONTER,LABEL=(4,NL),
//      DISP=OLD,DCB=(DEN=1,RECFM=U,BLKSIZE=32016)
```

```
//GO.FT04F001   DD   UNIT=2400,VOL=SER=NPS282,DSNAME=MONTER,LABEL=(1,SL),
//      DISP=(NEW,KEEP),DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016)
//GO.FT04F002   DD   UNIT=2400,VOL=SER=NPS282,DSNAME=MONTER,LABEL=(2,SL),
//      DISP=(NEW,KEEP),DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016)
//GO.FT04F003   DD   UNIT=2400,VOL=SER=NPS282,DSNAME=MONTER,LABEL=(3,SL),
//      DISP=(NEW,KEEP),DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016)
//GO.FT04F004   DD   UNIT=2400,VOL=SER=NPS282,DSNAME=MONTER,LABEL=(4,SL),
//      DISP=(NEW,KEEP),DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016)
```

PROGRAM NO. 3

```
//   EXEC FORTCLG,REGION.GO=245K
//FORT.SYSIN   DD *
C       MAIN PROGRAM - LINEAR SYSTEM IDENTIFICATION
C       PURPOSE
C          TO IDENTIFY LINEAR TIME INVARIANT SYSTEMS ON
C          THE BASIS OF INPUT-OUTPUT RECORDS
C       DESCRIPTION OF PARAMETERS
C          INPUT
C          NP      -ESTIMATED NUMBER OF POLES
C          NZ      -ESTIMATED NUMBER OF ZEROES
C          KPTMAX - NUMBER OF DATA POINTS
C          IPTS    -DATA POINTS INTEGRATED PER LINEAR EQ.
C          T       -TIME
C          R       -INPUT AMPLITUDE AT TIME T
C          C       -OUTPUT AMPLITUDE AT TIME T
C       REMARKS
C          (1) OUTPUT WILL CONSIST OF A TRANSFER FUNCTION
C              AND STATE VARIABLE REPRESENTATION OF SYSTEM
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C          (1) DLLSQ
C          (2) RTPLSB
C       METHOD
C          MULTIPLE INTEGRALS OF THE INPUT AND OUTPUT DATA
C          ARE USED TO FORMULATE A SET OF OVERDETERMIMED
C          LINEAR EQUATIONS.THESE EQUATIONS ARE THEN SOLVED
C          FOR THE UNKNOWN MODEL PARAMETERS USING THE METHOD
C          OF LEAST SQUARES.
C       ..............................................................
        DIMENSION DAT(8004)
        REAL*8 DT2
        REAL*8 A(2600),B(200),X(26),AUX(52),CONV(20)
        REAL*8 AP(20),AZ(20),PRA(20),PIA(20),ZRA(20),ZIA(20)
        REAL*8 TO,TN(20),RO(20),RN(20),CO(20),CN(20)
        REAL*8 CH1(4002),CH2(4002)
        INTEGER IPIV(26)
        REWIND 4
        KPTMAX=4002
        IPTS=100
        NUR=1
        NP=2
        NP=3
        NZ=0
      5 CONTINUE
```

```
   99 MEQS=KPTMAX/IPTS
      TO=0.0
      DT=0.0001
      EPS=10.0**(-35)
      K=1
      M=0
      NP1=NP+1
      NP2=NP+2
      NZ1=NZ+1
      N=NP+NZ+1
      NI=N+NP
C
C     PASS THE DATA FROM THE TAPE TO. THE VECTORS CH1(L),CH2(L)
C
      READ(4,8) DAT
    8 FORMAT(100(80A4),4A4)
      WRITE(6,9) DAT
    9 FORMAT(10F10.4)
C
C     SET CUMULATIVE INTEGRAL VALUES TO ZERO
C
      DO 10 I=2,NP2
      RO(I)=0.00
   10 CO(I)=0.00
      L=0
      DO 11 I=1,8004,2
      L=L+1
      CH1(L)=DAT(I)
   11 CH2(L)=DAT(I+1)
C
C     READ IN INITIAL DATA POINT (T,R,C)
C
      RO(1)=CH1(1)
      CO(1)=CH2(1)
      TOFF=TO
      TN(1)=TO
   15 K=K+1
      TN(1)=DT*FLOAT(K)
C
C     READ IN NEW DATA POINT
C
      RN(1)=CH1(K+1)
      CN(1)=CH2(K+1)
C
C     UPDATE MULTIPLE INTEGRATIONS
C
      DT2=(TN(1)-TO)*0.5
      DO 20 INT=1,NP1
```

```
      RN(INT+1)=(RO(INT)+RN(INT))*DT2+RO(INT+1)
   20 CN(INT+1)=(CO(INT)+CN(INT))*DT2+CO(INT+1)
C
C     FORM A LINEAR EQUATION
C
      IF(K.NE.(K/IPTS)*IPTS) GO TO 35
      M=M+1
      B(M)=CN(2)
      TN(2)=(TN(1)-TOFF)
      DO 25 I=1,NP
      IA=(NP-I)*MEQS+M
      IC=(N+I-1)*MEQS+M
      TN(I+2)=TN(I+1)*TN(2)/FLOAT(I+1)
      A(IA)=-CN(I+2)
   25 A(IC)=TN(I+1)
      DO 30 I=1,NZ1
      IA=(NP+I-1)*MEQS+M
      IRN=NP+3-I
   30 A(IA)=RN(IRN)
C
C     RESET OLD VALUES
C
   35 TO=TN(1)
      DO 40 I=1,NP2
      RO(I)=RN(I)
   40 CO(I)=CN(I)
      IF(M.LT.MEQS) GO TO 15
C
C     SOLVE FOR PARAMETERS BY METHOD OF LEAST SQUARES
C
      CALL DLLSQ(A,B,M,NI,1,X,IPIV,EPS,IER,AUX)
C
C     CALCULATE POLES
C
      AP(1)=1.00
      DO 45 I=1,NP
      J=NP+2-I
   45 AP(J)=X(I)
      CALL RTPLSB(NP,AP,PRA,PIA,CONV,IERPZ)
C
C     CALCULATE ZEROES
C
      GAINI=X(N)
      DO 50 I=NP1,N
   50 X(I)=X(I)/X(N)
      DO 55 I=1,NZ1
      J=N+1-I
   55 AZ(I)=X(J)
```

```
      IF(NZ.EQ.0) GO TO 60
      CALL RTPLSB(NZ,AZ,ZRA,ZIA,CONV,IERPZ)
   60 CONTINUE
C
C     OUTPUT
C
      WRITE(6,914)
      WRITE(6,915) K
      WRITE(6,916) M
      WRITE(6,917) IPTS
      WRITE(6,918) EPS
      WRITE(6,919) AUX (1)
      WRITE(6,920) IER
      WRITE(6,921) (IPIV(I),I=1,N)
      WRITE(6,900)
      WRITE(6,901)
      WRITE(6,902)
      DO 65  I=1,NP
      WRITE(6,903) I,PRA(I),PIA(I)
   65 CONTINUE
      WRITE(6,904)
      IF(NZ.EQ.0) GO TO 75
      DO 70  I=1,NZ
      WRITE(6,903) I,ZRA(I),ZIA(I)
   70 CONTINUE
   75 CONTINUE
      WRITE(6,905) GAINI
      WRITE(6,906)
      WRITE(6,907)
      DO 80  I=1,NP
      II=NP+2-I
      WRITE(6,910) I,AP(II)
   80 CONTINUE
      WRITE(6,908)
      WRITE(6,910) NP,GAINI
      WRITE(6,909)
      DO 85  I=1,NZ1
      II=NZ+2-I
      WRITE(6,910) I,AZ(II)
   85 CONTINUE
      WRITE(6,913)
      DO 90  I=1,NP
      II=N+I
      WRITE(6,910) I,X(II)
   90 CONTINUE
      NUR=NUR+1
      IF(NUR.GT.4) GO TO 88
      READ(4,8,END=99) DAT
```

```
900 FORMAT(1H1,///,25X,'IDENTIFICATION OF UNKNOWN SYSTEM')
901 FORMAT(///,12X,'SYSTEM TRANSFER FUNCTION')
902 FORMAT(//,15X,'POLES',12X,'REAL',13X,'INAGINARY',/)
903 FORMAT(17X,I2,7X,G15.8,6X,G15.8,1X,'J',/)
904 FORMAT(//,15X,'ZEROES',11X,'REAL',13X,'IMAGINARY',/)
905 FORMAT(//,15X,'GAIN CONSTANT =',G15.8,/)
906 FORMAT(////,12X,'SYSTEM STATE VARIABLES (PHASE FORM)')
907 FORMAT(//,15X,'A VECTOR',/)
908 FORMAT(//,15X,'B VECTOR',/)
909 FORMAT(//,15X,'C VECTOR',/)
910 FORMAT(17X,I2,7X,G15.8,/)
911 FORMAT(1H1,9X,'IER',I5,5X,'EPS',E15.8,5X,'AUX',
   1E16.8,/,9X,11I8)
912 FORMAT(//,5X,4(E20.9,5X),//,5X,4(E20.9,5X),//)
913 FORMAT(//,15X,'INITIAL STATE VECTOR',/)
914 FORMAT(//////,12X,'PROGRAM PARAMETERS',/)
915 FORMAT(15X,'NUMBER OF DATA POINTS     =',I6,/)
916 FORMAT(15X,'NUMBER OF EQUATIONS       =',I6,/)
917 FORMAT(15X,'DATA POINTS PER EQUATION =',I6,/)
918 FORMAT(15X,'EPS        =',E16.8,/)
919 FORMAT(15X,'RMS ERROR =',E16.8,/)
920 FORMAT(15X,'IER        =',I4,/)
921 FORMAT(15X,'IPIV(I)    =',15(I2,1X),/)
 88 STOP
    END
C
C        SUBROUTINE DLLSQ
C
C        PURPOSE
C           TO SOLVE LINEAR LEAST SQUARES PROBLEMS, I.E. TO MINIMIZE
C           THE EUCLIDEAN NORM OF B-A*X, WHERE A IS A M BY N MATRIX
C           WITH M NOT LESS THAN N. IN THE SPECIAL CASE M=N SYSTEMS OF
C           LINEAR EQUATIONS MAY BE SOLVED.
C
C        USAGE
C           CALL DLLSQ (A,B,M,N,L,X,IPIV,EPS,IER,AUX)
C
C        DESCRIPTION OF PARAMETERS
C           A      - DOUBLE PRECISION M BY N COEFFICIENT MATRIX
C                    (DESTROYED).
C           B      - DOUBLE PRECISION M BY L RIGHT HAND SIDE MATRIX
C                    (DESTROYED).
C           M      - ROW NUMBER OF MATRICES A AND B.
C           N      - COLUMN NUMBER OF MATRIX A, ROW NUMBER OF MATRIX X.
C           L      - COLUMN NUMBER OF MATRICES B AND X.
C           X      - DOUBLE PRECISION N BY L SOLUTION MATRIX.
C           IPIV   - INTEGER OUTPUT VECTOR OF DIMENSION N WHICH
C                    CONTAINS INFORMATIONS ON COLUMN INTERCHANGES
```

```
C                IN MATRIX A. (SEE REMARK NO.3).                          DLLS 260
C        EPS   - SINGLE PRECISION INPUT PARAMETER WHICH SPECIFIES         DLLS 270
C                A RELATIVE TOLERANCE FOR DETERMINATION OF RANK OF        DLLS 280
C                MATRIX A.                                                DLLS 290
C        IER   - A RESULTING ERROR PARAMETER.                             DLLS 300
C        AUX   - A DOUBLE PRECISION AUXILIARY STORAGE ARRAY OF            DLLS 310
C                DIMENSION MAX(2*N,L). ON RETURN FIRST L LOCATIONS        DLLS 320
C                OF AUX CONTAIN THE RESULTING LEAST SQUARES.              DLLS 330
C                                                                         DLLS 340
C     REMARKS                                                             DLLS 350
C        (1) NO ACTION BESIDES ERROR MESSAGE IER=-2 IN CASE               DLLS 360
C            M LESS THAN N.                                               DLLS 370
C        (2) NO ACTION BESIDES ERROR MESSAGE IER=-1 IN CASE               DLLS 380
C            OF A ZERO-MATRIX A.                                          DLLS 390
C        (3) IF RANK K OF MATRIX A IS FOUND TO BE LESS THAN N BUT         DLLS 400
C            GREATER THAN 0, THE PROCEDURE RETURNS WITH ERROR CODE        DLLS 410
C            IER=K INTO CALLING PROGRAM. THE LAST N-K ELEMENTS OF         DLLS 420
C            VECTOR IPIV DENOTE THE USELESS COLUMNS IN MATRIX A.          DLLS 430
C            THE REMAINING USEFUL COLUMNS FORM A BASE OF MATRIX A.        DLLS 440
C        (4) IF THE PROCEDURE WAS SUCCESSFUL, ERROR PARAMETER IER         DLLS 450
C            IS SET TO 0.                                                 DLLS 460
C                                                                         DLLS 470
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       DLLS 480
C        NONE                                                             DLLS 490
C
C     METHOD
C        HOUSEHOLDER TRANSFORMATIONS ARE USED TO TRANSFORM MATRIX A       DLLS 520
C        TO UPPER TRIANGULAR FORM. AFTER HAVING APPLIED THE SAME          DLLS 530
C        TRANSFORMATION TO THE RIGHT HAND SIDE MATRIX B, AN               DLLS 540
C        APPROXIMATE SOLUTION OF THE PROBLEM IS COMPUTED BY               DLLS 550
C        BACK SUBSTITUTION. FOR REFERENCE, SEE                            DLLS 560
C        G. GOLUB, NUMERICAL METHODS FOR SOLVING LINEAR LEAST             DLLS 570
C        SQUARES PROBLEMS, NUMERISCHE MATHEMATIK, VOL.7,                  DLLS 580
C        ISS.3 (1965), PP.206-216.                                        DLLS 590
C                                                                         DLLS 600
C     ..................................................................DLLS 610
C                                                                         DLLS 620

      SUBROUTINE DLLSQ(A,B,M,N,L,X,IPIV,EPS,IER,AUX)                      DLLS 630
C                                                                         DLLS 640
      DIMENSION A(1),B(1),X(1),IPIV(1),AUX(1)                             DLLS 650
      DOUBLE PRECISION A,B,X,AUX,PIV,H,SIG,BETA,TOL                       DLLS 660
C                                                                         DLLS 670
C     ERROR TEST                                                          DLLS 680
      IF(M-N)30,1,1                                                       DLLS 690
C                                                                         DLLS 700
C     GENERATION OF INITIAL VECTOR S(K) (K=1,2,...,N) IN STORAGE          DLLS 710
```

```
C        LOCATIONS AUX(K) (K=1,2,...,N)                                   DLLS 720
       1 PIV=0.D0                                                         DLLS 730
         IEND=0                                                           DLLS 740
         DO 4 K=1,N                                                       DLLS 750
         IPIV(K)=K                                                        DLLS 760
         H=0.D0                                                           DLLS 770
         IST=IEND+1                                                       DLLS 780
         IEND=IEND+M                                                      DLLS 790
         DO 2 I=IST,IEND                                                  DLLS 800
       2 H=H+A(I)*A(I)                                                    DLLS 810
         AUX(K)=H                                                         DLLS 820
         IF(H-PIV)4,4,3                                                   DLLS 830
       3 PIV=H                                                            DLLS 840
         KPIV=K                                                           DLLS 850
       4 CONTINUE                                                         DLLS 860
C                                                                         DLLS 870
C        ERROR TEST                                                       DLLS 880
         IF(PIV)31,31,5                                                   DLLS 890
C                                                                         DLLS 900
C        DEFINE TOLERANCE FOR CHECKING RANK OF A                          DLLS 910
       5 SIG=DSQRT(PIV)                                                   DLLS 920
         TOL=SIG*ABS(EPS)                                                 DLLS 930
C                                                                         DLLS 940
C                                                                         DLLS 950
C        DECOMPOSITION LOOP                                               DLLS 960
         LM=L*M                                                           DLLS 970
         IST=-M                                                           DLLS 980
         DO 21 K=1,N                                                      DLLS 990
         IST=IST+M+1                                                      DLLS1000
         IEND=IST+M-K                                                     DLLS1010
         I=KPIV-K                                                         DLLS1020
         IF(I)8,8,6                                                       DLLS1030
C                                                                         DLLS1040
C        INTERCHANGE K-TH COLUMN OF A WITH KPIV-TH IN CASE KPIV.GT.K      DLLS1050
       6 H=AUX(K)                                                         DLLS1060
         AUX(K)=AUX(KPIV)                                                 DLLS1070
         AUX(KPIV)=H                                                      DLLS1080
         ID=I*M                                                           DLLS1090
         DO 7 I=IST,IEND                                                  DLLS1100
         J=I+ID                                                           DLLS1110
         H=A(I)                                                           DLLS1120
         A(I)=A(J)                                                        DLLS1130
       7 A(J)=H                                                           DLLS1140
C                                                                         DLLS1150
C        COMPUTATION OF PARAMETER SIG                                     DLLS1160
       8 IF(K-1)11,11,9                                                   DLLS1170
       9 SIG=0.D0                                                         DLLS1180
         DO 10 I=IST,IEND                                                 DLLS1190
```

```
   10 SIG=SIG+A(I)*A(I)                                          DLLS1200
      SIG=DSQRT(SIG)                                             DLLS1210
C                                                                DLLS1220
C     TEST ON SINGULARITY                                        DLLS1230
      IF(SIG-TOL)32,32,11                                        DLLS1240
C                                                                DLLS1250
C     GENERATE CORRECT SIGN OF PARAMETER SIG                     DLLS1260
   11 H=A(IST)                                                   DLLS1270
      IF(H)12,13,13                                              DLLS1280
   12 SIG=-SIG                                                   DLLS1290
C                                                                DLLS1300
C     SAVE INTERCHANGE INFORMATION                               DLLS1310
   13 IPIV(KPIV)=IPIV(K)                                         DLLS1320
      IPIV(K)=KPIV                                               DLLS1330
C                                                                DLLS1340
C     GENERATION OF VECTOR UK IN K-TH COLUMN OF MATRIX A AND OF  DLLS1350
C     PARAMETER BETA                                             DLLS1360
      BETA=H+SIG                                                 DLLS1370
      A(IST)=BETA                                                DLLS1380
      BETA=1.DO/(SIG*BETA)                                       DLLS1390
      J=N+K                                                      DLLS1400
      AUX(J)=-SIG                                                DLLS1410
      IF(K-N)14,19,19                                            DLLS1420
C                                                                DLLS1430
C     TRANSFORMATION OF MATRIX A                                 DLLS1440
   14 PIV=0.DO                                                   DLLS1450
      ID=0                                                       DLLS1460
      JST=K+1                                                    DLLS1470
      KPIV=JST                                                   DLLS1480
      DO 18 J=JST,N                                              DLLS1490
      ID=ID+M                                                    DLLS1500
      H=0.DO                                                     DLLS1510
      DO 15 I=IST,IEND                                           DLLS1520
      II=I+ID                                                    DLLS1530
   15 H=H+A(I)*A(II)                                             DLLS1540
      H=BETA*H                                                   DLLS1550
      DO 16 I=IST,IEND                                           DLLS1560
      II=I+ID                                                    DLLS1570
   16 A(II)=A(II)-A(I)*H                                         DLLS1580
C                                                                DLLS1590
C     UPDATING OF ELEMENT S(J) STORED IN LOCATION AUX(J)         DLLS1600
      II=IST+ID                                                  DLLS1610
      H=AUX(J)-A(II)*A(II)                                       DLLS1620
      AUX(J)=H                                                   DLLS1630
      IF(H-PIV)18,18,17                                          DLLS1640
   17 PIV=H                                                      DLLS1650
      KPIV=J                                                     DLLS1660
   18 CONTINUE                                                   DLLS1670
```

```
C                                                                       DLLS1680
C     TRANSFORMATION OF RIGHT HAND SIDE MATRIX B                        DLLS1690
   19 DO 21 J=K,LM,M                                                    DLLS1700
      H=0.D0                                                            DLLS1710
      IEND=J+M-K                                                        DLLS1720
      II=IST                                                            DLLS1730
      DO 20 I=J,IEND                                                    DLLS1740
      H=H+A(II)*B(I)                                                    DLLS1750
   20 II=II+1                                                           DLLS1760
      H=BETA*H                                                          DLLS1770
      II=IST                                                            DLLS1780
      DO 21 I=J,IEND                                                    DLLS1790
      B(I)=B(I)-A(II)*H                                                 DLLS1800
   21 II=II+1                                                           DLLS1810
C     END OF DECOMPOSITION LOOP                                         DLLS1820
C                                                                       DLLS1830
C                                                                       DLLS1840
C     BACK SUBSTITUTION AND BACK INTERCHANGE                            DLLS1850
      IER=0                                                             DLLS1860
      I=N                                                               DLLS1870
      LN=L*N                                                            DLLS1880
      PIV=1.D0/AUX(2*N)                                                 DLLS1890
      DO 22 K=N,LN,N                                                    DLLS1900
      X(K)=PIV*B(I)                                                     DLLS1910
   22 I=I+M                                                             DLLS1920
      IF(N-1)26,26,23                                                   DLLS1930
   23 JST=(N-1)*M+N                                                     DLLS1940
      DO 25 J=2,N                                                       DLLS1950
      JST=JST-M-1                                                       DLLS1960
      K=N+N+1-J                                                         DLLS1970
      PIV=1.D0/AUX(K)                                                   DLLS1980
      KST=K-N                                                           DLLS1990
      ID=IPIV(KST)-KST                                                  DLLS2000
      IST=2-J                                                           DLLS2010
      DO 25 K=1,L                                                       DLLS2020
      H=B(KST)                                                          DLLS2030
      IST=IST+N                                                         DLLS2040
      IEND=IST+J-2                                                      DLLS2050
      II=JST                                                            DLLS2060
      DO 24 I=IST,IEND                                                  DLLS2070
      II=II+M                                                           DLLS2080
   24 H=H-A(II)*X(I)                                                    DLLS2090
      I=IST-1                                                           DLLS2100
      II=I+ID                                                           DLLS2110
      X(I)=X(II)                                                        DLLS2120
      X(II)=PIV*H                                                       DLLS2130
   25 KST=KST+M                                                         DLLS2140
C                                                                       DLLS2150
```

```
C
C     COMPUTATION OF LEAST SQUARES
   26 IST=N+1
      IEND=0
      DO 29 J=1,L
      IEND=IEND+M
      H=0.D0
      IF(M-N)29,29,27
   27 DO 28 I=IST,IEND
   28 H=H+B(I)*B(I)
      IST=IST+M
   29 AUX(J)=H
      RETURN
C
C     ERROR RETURN IN CASE M LESS THAN N
   30 IER=-2
      RETURN
C
C     ERROR RETURN IN CASE OF ZERO-MATRIX A
   31 IER=-1
      RETURN
C
C     ERROR RETURN IN CASE OF RANK OF MATRIX A LESS THAN N
   32 IER=K-1
      RETURN
      END
//GO.FT06F001 DD SPACE=(CYL,(3,1))
//GO.FT04F001  DD  UNIT=2400,VOL=SER=NPS330,DSNAME=MONTER,LABEL=(1,SL),
//       DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016),DISP=OLD
//GO.FT04F002  DD  UNIT=2400,VOL=SER=NPS330,DSNAME=MONTER,LABEL=(2,SL),
//       DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016),DISP=OLD
//GO.FT04F003  DD  UNIT=2400,VOL=SER=NPS330,DSNAME=MONTER,LABEL=(3,SL),
//       DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016),DISP=OLD
//GO.FT04F004  DD  UNIT=2400,VOL=SER=NPS330,DSNAME=MONTER,LABEL=(4,SL),
//       DCB=(DEN=2,RECFM=FB,LRECL=32016,BLKSIZE=32016),DISP=OLD
```

PROGRAM NO. 4

```
THIS PROGRAM SOLVES A SET OF LINEAR DIFFERENTIAL EQUETIONS
THE SOLUTION APPEARS AS A DRAW

//   EXEC FORTCLGP,PARM.FORT='LIST,MAP',REGION.GO=120K
//FORT.SYSPRINT  DD  SYSOUT=A,SPACE=(CYL,6)
      DIMENSION XX(900),YY(900),ZZ(900),ZX(900),ZY(900)
      REAL*8 X(20),XDOT(20),ITITLE(12)/'MONTER-1',11*'        '/,T,DT
      REAL*4 LAB/'TEST'/,ALAB/'TES5'/,BLAB/'TES3'/,CLAB/'TES2'/
      DO 1 I=1,13
    1 X(I)=0.D0
      T=0.
      DT=0.01
      NT=0
C
C     SET DIF. EQUE. IN STATE FORM
C
      DO 7 I=1,900
    2 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
      XDOT(4)=1.-33000.*X(1)-37630.*X(2)-4773.*X(3)-144.*X(4)
      XDOT(5)=X(6)
      XDOT(6)=X(7)
      XDOT(7)=X(8)
      XDOT(8)=1.-42867.6*X(5)-47640.8*X(6)-5600.3*X(7)-155.9*X(8)
      XDOT(9)=X(10)
      XDOT(10)=X(11)
      XDOT(11)=1.-6785.3*X(9)-2500.4*X(10)-65.1*X(11)
      XDOT(12)=X(13)
      XDOT(13)=1.-126.*X(12)-40.1*X(13)
C
C   THIS SUBROUTINE SOLVES DIFF. EQUET.
C
      S=RKLDEQ(13,X,XDOT,T,DT,NT)
      IF(S-1.) 5,2,6
C
C    STOP IF INTEGRATION TROUBLE OCCURS
C
    5 WRITE(6,4)
    4 FORMAT(//T8,'INTEGRATION TROUBLE')
      GO TO 9
```

```
    6 TT=I
C
C     OUTPUTS
C
      XX(I)=TT*DT
      YY(I)=52500.*(5.*X(1)+X(2))
      ZZ(I)=64711.8*(5.*X(5)+X(6))
      ZX(I)=28337.5*X(9)
      ZY(I)=537.3*X(12)
    7 CONTINUE
C
C   SUBROUTINE DRAW , DRAWS OUTPUTS
C
      CALL DRAW(900,XX,YY,1,0,LAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
      CALL DRAW(900,XX,ZZ,2,0,ALAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
      CALL DRAW(900,XX,ZX,2,0,BLAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
      CALL DRAW(900,XX,ZY,3,0,CLAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
    8 FORMAT(I6)
    9 STOP
      END
```

BIBLIOGRAPHY

1. Nieman, R. E., Fisher, D. G., and Seborg, D. E., "A Review of Process Identification and Parameter Estimation Techniques," International Journal of Control, V. 13, No. 2, p. 209-264, May 1971.

2. Eykhoff, P., van der Grinten, M., Kwakernaai, H., and Veltman, B., "Survey-Systems Modeling and Identification," Third Congress of I.F.A.C., London, June 1966.

3. Mishkin, E., and Haddad, R. A., "Identification and Command Problems in Adaptive Systems, "IRE Trans. on Automatic Control, V. AC-4, #2, p. 121-131, November 1959.

4. Levin, M. J., "Optimum Estimation of Impulse Response in the Presence of Noise," IRE Trans. on Circuit Theory, V.

5. Kerr, R. B., and Surber, Jr., N. H., "Precision of Impulse Response Identification Based on Short, Normal Operating Records," IRE Trans. on Automatic Control, V. AC-6, No. 2, p. 173-182, May 1961.

6. Turin, G. L., "On the Estimation in the Presence of Noise of the Impulse Response of a Random, Linear Filter," IRE Trans. on Information Theory, V. IT-3, No. 1, p. 5-10, March 1957.

7. Lichtenberger, W. W., "On the Identification of Linear Processes," IRE Trans. on Circuit Theory, v. CT-7, no. 1, p. 78-79, March 1960.

8. Lichtenberger, W.W., "A Technique of Linear System Identification Using Correlating Filters," IRE Trans. on Automatic Control, v. AC-9, no. 2, pp. 183-199, May 1961.

9. Hill, J.D., and McMurtry, "An Application of Digital Computers to Linear System Idnetification," IEEE Trans. on Automatic Control, v. AC-9, no. 5, pp. 536-538, October 1964.

10. Lindenlaub, J.C., and Cooper, G.R., "Noise Limitations of System Identification Techniques," IEEE Trans. on Automatic Control, v. AC-10, no. 1, pp. 43-48, January 1963.

11. Kumar, K.S.P., and Sridhar, R., "On the Identification of Control Systems by the Quasi-Linearization Method," IEEE Trans. on Automatic Control, v. AC-9 no. 2, pp. 151-154, April 1964.

12. Nagumo, J., and Noda, A., "A Learning Method for System Identification," IEEE Trans. on Automatic Control, v. AC-12, no. 3, pp. 282-287, June 1967.

13. Bass, R., Testing Function Techniques in System Identification, Ph.D., Dissertation, University of California, Berkeley, March 1969.

14. Astrom, K.J., and Bohlin, T., "Numerical Identification of Linear Dynamic Systems from Normal Operating Records," IFAC Symposium - Theory of Self-Adaptive Systems, Teddington, 1965.

15. Peterka, V., "New Approach to the Identification of Discrete Dynamic Systems," Kybernetika, vol. 4, no. 5, 1968.

16. Peterka, V., and Smuk, K., "On-Line Estimation of Dynamic Model Parameters from Input-Output Data," IFAC Congress, Warsaw, 1969.

17. Ho, B.L., and Kalman, R.E., "Effective Construction of Linear State Variable Models from Input/Output Functions," Sonderdruck aus Regelungstechnik, v. 14, no. 12, pp. 545-548, 1966.

18. Eldem, I., Identification of Plant Dynamics Using Ho's Algorithm, MS Thesis, United States Naval Postgraduate School, Monterey, California, 1969.

19. Diamesis, J.E., "A New Method for Determining the Parameters of Physical Systems, Proc. of the IEEE, v. 53, no. 2, pp. 205-206, February 1965.

20. Hansell, W.R., "Identification of System Dynamics Using Multiple Integration," Thesis, M.S., United States Naval Postgraduate School, Monterey, California, 1969.

21. Raney, S.D., "Procedures for Converting 7-track Magnetic Tapes to 9-track Magnetic Tapes," Technical Note No. 0211-08, Second Edition, Computer Center, United States Naval Postgraduate School, Monterey, California.

22. Golub, G., "Numerical Methods for Solving Linear Least Squares Problems," Numerische Mathematik, v. 7, pp. 206-216, 1965.

23. Businger, P., and Golub, G., "Lineary Least Squares Solutions by Householder Transformations," Numerische Mathematik, v. 7, pp. 269-276, 1965.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
2.	Professor George J. Thaler, Code 52Tr Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	5
3.	Professor Harold Titus, Code 52Ts Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
4.	Professor Ralph Bach, Code 52Bz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
5.	Mr. James W. Banham Head, Machinery Automation Systems Department Naval Ship Engineering Center Philadelphia Division Philadelphia, Pennsylvania 19112	1
6.	Mr. James R. Garnett Project Engineer, Systems Analysis Branch Naval Ships Engineering Center Philadelphia Division Philadelphia, Pennsylvania 19112	1
7.	Mr. Joseph Rubis Naval Ships Research & Development Center Annapolis, Maryland 21402	1
8.	Mr. Walt Blumberg Naval Ships Research & Development Center Annapolis, Maryland 21402	1
9.	Professor Denis Towill University of Wales Institute of Sciences & Technology Cardiff, Wales	1

No. Copies

10. LTJG W. R. Hansell, USN 1
257 10th Street
Idaho Falls, Idaho 83401

11. Excmo. Sr. Almirante Director de 1
Ensenanza Naval
Ministerio de Marina
Madrid, Spain

12. Ilmo. Sr. C. de N. Director ETEA 1
E. T. E. A.
Vigo, Spain

13. Lt. Angel J. Montero, Spanish Navy 1
E. T. E. A.
Vigo, Spain

14. Defense Documentation Center 2
Cameron Station
Alexandria, Virginia 22314

15. Professor Javier Aracil 1
Escuela Tecnica Superior
Ingenieros Industriales
Avenida Reina Mercedes
Sevilla, Spain

16. Professor Juan Peracalla 1
Escuela Technica Superior
Ingenieros Industriales
Departamento de Electronica
Avenida Generalisimo Franco 999
Barcelona 14, Spain

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Identification of System Dynamics Using Multiple Integrations, Tests of Physical Systems.

4. DESCRIPTIVE NOTES (Type of report and, inclusive dates)

Master's Thesis; June 1972

5. AUTHOR(S) (First name, middle initial, last name)

Angel Jose Montero

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1972	153	23

8a. CONTRACT OR GRANT NO. b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

A method for identifying and - after identification - reducing linear time invariant systems on the basis of input - output records of the system is reviewed, applied to physical devices and extended to handle the case where noise is present. The method is implemented using an analog to digital converter in order to have the input and output records in a digitized form and thus to be able to use a digital computer program composed of a numerical integration subroutine and another subroutine to solve overdetermined sets of linear algebraic equations. Several practical examples are presented to demonstrate the accuracy and present capabilities of the procedure.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-314

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Linear Systems						
Identification						
Multiple Integrals						
Modeling						
Reduction						

135433

Thesis
M689
c.1

Montero
Identification of
system dynamics using
multiple integrations,
tests of physical sys-
tems.